Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

DEVELOPMENT OF DEEP LEARNING NEURAL NETWORK FOR ECOLOGY DATA AND MEDICAL IMAGE

by
Shaobo Liu

Deep learning in computer vision and image processing has attracted attentions from various fields including ecology and medical image. Ecologists are interested in finding an effective model structure to classify different species. Tradition deep learning model use a convolutional neural network, such as LeNet, AlexNet, VGG models, residual neural network, and inception models, are first used on classifying bee wing and butterfly datasets. However, insufficient data sample and unbalanced samples in each class have caused a poor accuracy. To make improvement the test accuracy, data augmentation and transfer learning are applied. Recently developed deep learning framework based on mathematical morphology also shows its effective in shape representation, contour detection and image smoothing. The experimental results in the morphological neural network shows this type of deep learning model is also effective in ecology datasets and medical dataset. Compared with CNN, the MNN could achieve a similar or better result in the following datasets.

The chest X-ray images are notoriously difficult to analyze for the radiologists due to their noisy nature. The existing models based on convolutional neural networks contain a giant number of parameters and thus require multi-advanced GPUs to deploy. In this research, the morphological neural networks are developed to classify chest X-ray images, including the Pneumonia Dataset and the COVID-19 Dataset. A novel structure, which can self-learn a morphological dilation or erosion, is proposed for determining the most suitable depth of the adaptive layer. Experimental results on the chest X-ray dataset and the

COVID-19 dataset show that the proposed model achieves the highest classification rate as comparing against the existing models. More significant improvement is that the proposed model reduces around 97% computational parameters of the existing models.

Automatic identification of pneumonia on medical images has attracted intensive studies recently. The model for detecting pneumonia requires both a precise classification model and a localization model. A joint-task joint learning model with shared parameters is proposed to combine the classification model and segmentation model. To accurately classify and localize pneumonia area. Experimental results using the massive dataset of Radiology Society of North America have confirmed the efficiency of showing a test mean interception over union (IoU) of 89.27% and a mean precision of area detection result of 58.45% in segmentation model. Then, two new models are proposed to improve the performance of the original joint-task learning model. Two new modules are developed to improve both classification and segmentation accuracies in the first model. These modules including an image preprocessing module and an attention module. In the second model, a novel design is used to combine both convolutional layers and morphological layers with an attention mechanism. Experimental results performed on the massive dataset of the Radiology Society of North America have confirmed its superiority over other existing methods. The classification test accuracy is improved from 0.89 to 0.95, and the segmentation model achieves an improved mean precision result from 0.58 to 0.78. Finally, two weakly-supervised learning methods: class-saliency map and grad-cam, are used to highlight corresponding pixels or areas which have significant influence on the classification model, such that the refined segmentation can focus on the correct areas with high confidence.

DEVELOPMENT OF DEEP LEARNING NEURAL NETWORK FOR ECOLOGY DATA AND MEDICAL IMAGE

by
Shaobo Liu

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Science

Department of Computer Science

May 2021

APPROVAL PAGE

DEVELOPMENT OF DEEP LEARNING NEURAL NETWORK FOR ECOLOGY DATA AND MEDICAL IMAGE

Shaobo Liu

Dr. Frank Y. Shih, Dissertation Advisor — Date
Professor of Computer Science, NJIT

Dr. Xiaoning Ding, Committee Member — Date
Associate Professor of Computer Science, NJIT

Dr. Zhi Wei, Committee Member — Date
Professor of Computer Science, NJIT

Dr. Gareth Russell, Committee Member — Date
Associate Professor of Biological Sciences, NJIT

Dr. Hai Phan, Committee Member — Date
Assistant Professor of Information Systems, NJIT

BIOGRAPHICAL SKETCH

Author: Shaobo Liu

Degree: Doctor of Philosophy

Date: May 2021

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Science,
  New Jersey Institute of Technology, Newark, NJ, 2021

- Master of Science in Electrical Engineering,
  Stevens Institute of Technology, Hoboken, NJ, 2016

- Bachelor of Science in Electrical Power Engineering,
  Agricultural University of Hebei, Baoding, P. R. China, 2014

Major: Computer Science

Presentations and Publications:

S. Liu, F. Y. Shih, G. Russell, K. Russell, and H. Phan, "Classification of ecological data by deep learning," *Pattern Recognition and Artificial Intelligence*, vol. 34, no. 13 pp. 2052010 (20 pages), Dec. 2020.

S. Liu, X. Zhong, and F. Y. Shih, "Joint learning for pneumonia classification and segmentation on medical images," *Pattern Recognition and Artificial Intelligence*, vol. 35, no. 5, pp. 2157003 (19 pages), May 2021.

S. Liu, F. Y. Shih, and X. Zhong, "Classification of chest X-ray images using novel adaptive morphological neural networks," *Pattern Recognition and Artificial Intelligence*, accepted.

谨以此文献给我的家人和朋友们

To My Beloved Family and Friends

ACKNOWLEDGMENT

First, I would like to express my sincere gratitude to my dissertation advisor Prof. Frank Y. Shih, my thesis advisor, for his advice, patience and supports in guiding my study and research in this fantastic area. His guidance and encouragements have help me all the time in doing research and writing this dissertation.

I am also thankful for the committee members, Professor Russel, Prof. Hai Phan, Prof. Zhi Wei and Pro. Xiaoning Ding for providing very useful advices and supports. They gave me great help on doing research and how to make my thesis work better.

I would like to thank my classmates: Hao Liu, Yin Xin, Meiyan Xie, Han Hu and Yahui Wang. They are the friends I meet in NJIT and I really learnt a lot form these fantastic people with great ideas. I also want to thank my lab mates: Xin Zhong, Yucong Shen and Yanan Yang for giving me great help and many valuable suggestions in model design and experiments.

I would like to thank the Department of Computer Science the for Teaching Assistanceship during my PH.D. period.

Most importantly, I heartly appreciate my family members: Shuchuan Liu, Kuan Li, Peiying Peng and Jinbo Li for their support and understanding throughout my life. Particularly, I am grateful to my wife, Zhi Li, for her love, support and encouragement.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF TABLES
(Continued)

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1
INTRODUCTION

1.1 Objective

The objective of this dissertation is to present applications of deep learning models for small datasets such as ecology datasets and medical datasets. First, traditional convolutional neural network, the Convolutional Neural Network, is applied to the ecology dataset, such as the bee wing dataset and the butterfly dataset. Since the capacity of the original dataset is a relatively small dataset, several measures are used to improve the CNN models' performance, such as data augmentation and transfer learning methods.

Second, a new deep learning model use a novel feature extraction mechanism, the morphology neural network, is applied to the ecological dataset and the medical images, such as chest X-ray images and Covid-19 dataset. The experimental results shows MNN can extract the features with relatively less parameters then the CNN models and achieves a relatively higher classification rate.

However, the drawbacks for MNN are also shown in experiments. For image like dogs and cats, which shares similar features, MNN will show a relatively lower classification accuracy.

To overcome the drawback for MNN models, a new model is proposed and presented. It overcomes previous difficulties and also reduced the model's parameters tremendously. Finally, a joint task learning model use the proposed structure and applied to medical images.

1.2 Background Information

Deep learning has recently received lots of attentions in various fields of pattern recognition. Deep learning, also called deep structured learning, is a broader kind of machine learning methods based on a large amount of data. Different from traditional machine learning methods, deep learning does not require domain experts' help in building feature extractors. As a part of machine learning, deep learning can be categorized into supervised or unsupervised learning. Deep learning can be applied for various tasks with different types of data. For example, one can apply the Convolutional Neural Network (CNN) for image classification or the Recursive Neural Network (RNN) for language processing. In computer vision, CNN is an effective framework to recognize and classify multiple targets due to an auto feature extraction ability. Thanks to the expansional growth of computation ability, different structures of convolutional neural networks are developed, especially for image classification and objective detection.

The CNN models are designed to process multi-arrays, especially for image data or video. Although they were proposed by Yann LeCun in 1995 [1], the limitations of computing capacity and incomplete mathematical proof made deep learning difficult to be accepted by researchers. With the recent development of computing capacity, deep learning has much more great performance than the traditional machine learning methods on object classification, object detection, natural language processing, etc.

In 2012, Alex Krizhevsky developed AlexNet [2] based on LeNet proposed by Yann LeCun. The AlexNet has a complex structure; although there are only eight layers, it has millions of parameters in the whole model. It won the champion of

ImageNet competition in 2012, with the result of 15.4% test error. The network is made up of five convolution layers, including max-pooling layer, dropout layer, and three fully connected layers. In 2014, Google company, proposed a large CNN network, called GoogleNet [3], which has 22 layers and achieves the error rate of 6.7% on ImageNet competition. Its success proves that much deeper network and more convolution layers will have much better performance. Another network developed in 2014 is the VGG network [4], which has 19 layers. The VGG network keeps the network deep enough, and in the meantime, it keeps the network simple. In 2015, ResNet [5] proposed by Microsoft Research Asian achieved an incredible error rate of 3.6% on ImageNet competition. ResNet uses a residual block to avoid the problem of degradation: gradient disappearance in the back propagation. However, it takes two to three weeks to finish training on an 8-GPU machine. The CNN network has been applied by researchers in many fields, such as video classification [7] and NLP [8], to develop new deep learning networks such as AlphaGo [9] and Generative Adversarial Network [10].

There has seldom research on the combination of deep learning and ecology. Previously, the classification of ecological image data was applied by traditional machine learning methods, including random forest, artificial neural networks, support vector machines, and genetic algorithms [11-17]. Specifically, for recognizing bee wings, researchers have tried various methods machine learning methods including support vector machines, Naïve Bayes [18], k-nearest neighbors [19] and logistic classifier [20]. These methods are relatively effective experts before the popularity of CNN, but mainly focusing on extract features by domain experts. However, currently

biologists especially ecologists are showing their interests in building an efficient species recognition system by using deep learning neural networks, given the reason that convolutional neural networks' automatic feature extraction outstanding performance.

Schneider et al. [21] used RNN to classify different types of animals from trap camera data. Their result shows the test accuracy reaches 93%, which delivers that deep learning methods have a promising future in the ecological research. Different from the following tasks, this one is to recognize different species from limited and unbalanced datasets. These datasets include 19 classes of wings belonging to bees in New Jersey, 10 classes different butterflies from all over the world. In ecology, species are various, and one specie usually has different kinds of subspecies. This task requires a robust classification model to identify spice's class from given image data. Concerning the great progress having made by the Convolutional Neural Network model, especially the backpropagation applied in the training phase, CNN should be suitable for the classification task. Although given the fact that some of the samples are really hard to be distinguished by human's vision system.

One problem faced in training CNN models in our ecology datasets is the limitation in amount and highly imbalanced dataset. For example, in the dataset of bee-wings images data differs from osmiageorgica. With 9 images to bombusimpatiens with 132 images. In order to solve this problem, two methods are proposed to increase its performance. The first solution is data augmentation, which focus on enlarge the dataset based on current dataset and perform image processing operations such as rotation, skewing and shearing. The result for our data augmentation is the that the

training dataset are enraged to a balanced dataset and an improvement in overall accuracy and single class accuracy. The second solution is by transfer learning [22]. This technique utilizes the parameters of a well-trained CNN model and performed to ecology classification task. Several pre-trained models which already been trained on large dataset are applied in ecology dataset and improve the model performance**.**

In AlexNet [2], VGG models [4] and residual model [5], a fixed kernel size is used in convolution layer. In GoogleNet [3], a novel convolution block consists 4 different feature maps is termed as Inception modules. With this enriched feature maps, GoogleNet (or Inception v1, follow by Inception v2 [31], Inception v3 [23], Inception v4 [32]) won the ILSVRC (ImageNet Large Scale Visual Recognition Competition) at 2014. The high performance for inception modules attracts more and more attentions in this area.

Mathematical morphology has been used in effectively extracting object features, such as shapes, regions, edges, skeleton, and convex hull, which can improve the object representation and description [33, 34]. Similar to a mask used in the convolution operation, mathematical morphology needs a structure element to perform the operation on the image. Two essential operations are dilation and erosion, and other operations are different combinations. Dilation tends to enlarge objects, while erosion tends to shrink it. Another application for mathematical morphology is image pre-processing like morphological filtering [35].

Shih and Moh [36] proposed to implement morphological operations using programmable neural networks. Davidson and Hummer [37] presented morphological neural networks (MNN) with applications. Masci et al. [38] proposed a method using

counter harmonic mean for dilation and erosion in the deep learning framework. Shih et al. [39] proposed a morphological deep learning framework using smooth local minimum and local maximum to simulate erosion and dilation, respectively.

Radiologists use chest X-ray images to diagnose diseases in the lung area. However, these images are noisy and hard to analyze the diseases, such as bacteria pneumonia, virus pneumonia or healthy. Moreover, we apply our model to recognize possible samples of the recent COVID-19 pandemic cases. We use different morphological layers, including dilation, erosion, opening, closing, etc., combined with convolutional neural networks. It can help convolutional neural networks to refine the feature extraction process. Furthermore, we develop adaptive morphological layers for feature extraction, which can determine a suitable morphological operation and structure elements in the training process.

In the past few years, pneumonia has ranked as a top-ten cause of death in the United States of America. An effective automatic pneumonia identification system on medical images will help doctors to find and localize the pneumonia area. The requirements for this system are twofold. First, this system should be effective in classifying the pneumonia body from thousands of health bodies. Then this system should be able to localize the pneumonia area with a mask.

In this research, a joint-task learning model is designed for image classification and image segmentation with shared feature extraction blocks is firstly be presented. The dataset is highly unbalancing, with 8,900 patience and 20,000 healthy body. In this paper, we first propose a baseline model that learns image classification and segmentation simultaneously. Two algorithms of saliency map and Grad-CAM for

image classification model explanation are adopted. Secondly, an image preprocessing module and an attention module are applied to refine the baseline model. Experimental results show these modules can separately improve the performance of the joint-task learning model. However, when the following modules are combined, the unguided MNN layers change the gradient and cause the saliency map and Grad-CAM focusing on irreverent area. To overcome the problem, the attention module is applied to refine the feature maps between morphological layers in both channel-wise and spatial attention modules. The MBAM successfully helps the model to focus on the corresponding area with higher confidence. Furthermore, by combining the CNN layers and morphological layers in the same feature extraction layer, a new designed model is further proposed and achieved a higher performance.

CHAPTER 2
CLASSIFICATION OF ECOLOGICAL DATA USING DEEP LEARNING METHODS

2.1 Convolution Neural Networks

Deep learning [40], as a part of machine learning, requires a large amount data to train and evaluate its performance. In computer vision, convolutional neural network is first proposed by Yann LeCun [1] and has been populated since 2011 when AlexNet [2], the first deep neural network, is used to process a large amount of data classification problem and surprised the world by winning the champion of 2012 ImageNet Challenge. This community keeps growing till now. Before understanding the reasons that why the convolution neural network grows so fast, it is essential to understand how this model works. Since CNN models are based on a similar structure proposed by Dr. Yann LeCun and LeNet-5 is the first convolution neural network using this design, a detailed study on this structure is necessary.

Figure. 2.1 shows the structure of LeNet-5, which is first used for the classification of hand written digits. LeNet-5 is composed by several layers with different function. Similar to other machine learning models applied on image data, LeNet-5 needs a feature representation method to compress one (grayscale image) or three (RGB image) 2D matrices in to a kind of feature representation.

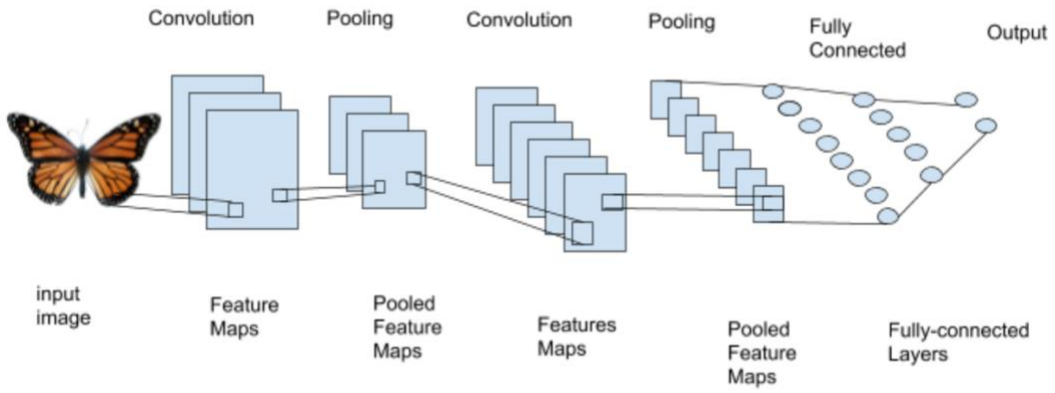


Figure 2.1 Structure of LeNet-5.

In LeCun's design, LeNet-5 contains an input layer which is used to read training or testing images. It is followed by a convolution layer used to extract features and a pooling layer used for reducing unnecessary data. After a second connection of convolutional layer with pooling layer, the feature representations are feed to a fully-connected artificial neural networks for classification.

In the convolutional layer, the input is one or several images with one or three channels, which could be grayscale or RGB images. In general, we perform convolution several times with different filters, so there are several output images, called feature maps. The convolutional layers extract different local features with different filters, making the whole network to learn all the main features in the input images. The convolutional layer followed by an activate function is described as:

$$h^k = f(\sum_{l \epsilon L} x^l \otimes w^k + b^k) \qquad (1.1)$$

where h^k is the latent representation of k-th feature map of the current layer, f is the activation function, x^l is the l-th feature map of group of feature maps L of the

previous layers or the l-th channel of the input images with totally L channels in the case of the first layer of the network, $\otimes$ denotes the 2D convolution operation, and w^k and b^k denote the weights (filters) and biases of the k-th feature map of the current layer respectively. A nonlinear function called ReLU (Rectified Linear Unit) works as the activation function f, which can be written as f(x) = max (0, x). This function will stay 0 when x is less than 0 but return to be x for any positive input. ReLU works well for neural network models because it allows the models to compute non-linearities and interaction, which makes ReLU a commonly used activation function.

Let a SoftMax function be defined as:

$$p_i = \frac{r^{z_i}}{\sum_{k=1}^{K} e^{z_k}}, \ i = 1, 2, 3, \dots, K \tag{1.2}$$

where z_i is an element of the input tensor. With SoftMax function, an N-dimensional vector of real numbers can be transfered into a vector of real numbers in range (0,1). The loss function is the cross-entropy , which is a widely-used alternative of squared error and defined as

$$H(y, p) = -\sum_i y_i log\ (p_i) \tag{1.3}$$

where y_i is the label of i-th input image and p_i is the i-th item of the output of SoftMax function.

The pooling layer is designed for perform down-sampling to image data. The purpose for down-sampling is to extract useful information and reduce the size of feature maps. Typically, there are two different down-sampling methods: average

pooling and max-pooling. Average pooling is used to compute the average value as feature in a small area and max-pooling is used to extract the maximum value in a small area.

After sufficient information is acquired from convolutional layers and pooling layers, the fully-connected layer is used to map the output to linearly separable space and flatten the matrix into a vector. Then SoftMax is used for regression to classify the data, so the output of the last fully-connected layer would be the predicted label.

AlexNet [2] is the first deep convolutional neural network. AlexNet is the first model to use ReLu as an activation function and utilize dropout layer. In ILSVRC 2010, AlexNet got the Top-1 and top-5 error rates of 37.5% and 17.0% respectively. An original design for AlexNet [2] is shown at Figure 2.2.

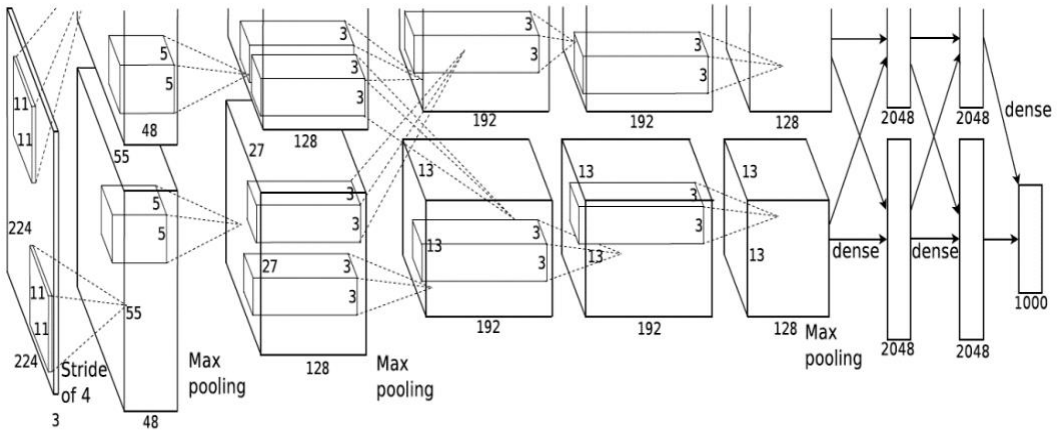


Figure 2.2 Design for AlexNet.

VGG neural network [4] is created by Visual Geometry Group. VGG-16 obtains 8.8% error rate and VGG-19 obtain 9.0% in ILSVRC 2014 (ImageNet Large Scale Visual Recognition Competition). With VGG19 stacked more convolutional layers than VGG16, the test error increased. Fig. 2.3 shows the structure of VGG16 & VGG19 model.

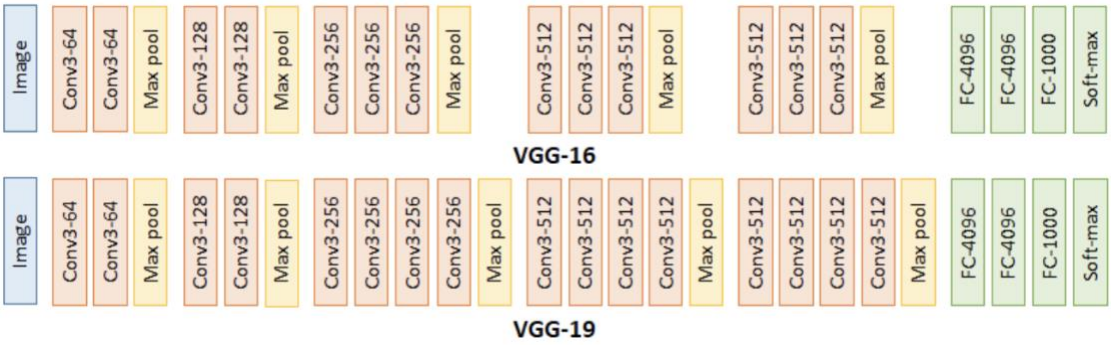


Figure 2.3 Structure [4] of VGG models.

VGG neural network [4] was developed by Visual Geometry Group, University of Oxford. In the 2014 ILSVRC (ImageNet Large Scale Visual Recognition Competition), VGG-16 obtained an error rate of 8.8% and VGG-19 obtained an error rate of 9.0%. In the VGG model, stacked convolution kernels with 3 by 3 are used. Note that two 3-by-3 convolution kernels equal to a 5-by-5 effective convolution area, three 3-by-3 kernels equal to a 7-by-7 effective area, and so on. The purpose of using stack convolutions is to reduce parameters in the learning process. The VGG16 contains two 5-by-5 convolutional layers and three 7-by-7 convolutional layers and the VGG19 contains two 5-by-5 convolutional layers and three 9-by-9 convolutional layers. However, when more convolution layers are stacked together, a vanishing gradient problem may happen. It is occurred during backpropagation when several

small derivatives are multiplied together after the same activation function. The problem of a small gradient will cause the parameters not to be updated effectively.

To solve the vanishing gradient problem, a new convolutional block, called residual block, is introduced in residual neural network [5]. By adding a shortcut connection between the input x to learn residual mapping $F(x)$ before the activation function, the output $x + F(x)$ can maintain a higher overall derivative. With residual connections, the residual neural network can add up to 152 layers. It won the competition in 2015 ILSVRC.

With a skip connection between activation functions, the problem of vanishing gradient problem in VGG model is solved. Fig. 2.4 shows the residual block in [5]. The shortcut connection is added between a short connection from input x to $F(x)$, the output H $(x) = x + F(x)$. The learnt residual mapping $F(x) = H(x) - x$. When $F(x)$ is close to 0, x can still pass to the next layer by shortcut connection. With residual connections, the residual can be added up to 152 layers.

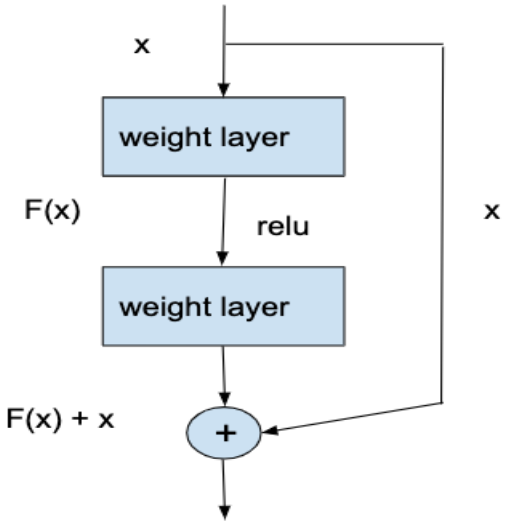


Figure 2.4 Residual block in Residual Network.

Inception block is first introduced by GoogleNet [3]. GoogleNet is also called Inception v1 and continued by Inception v2 [31], Inception v3 [23] and Inception v4 [32]. Inception v1 is the winner of the ILSVRC (ImageNet Large Scale Visual Recognition Competition) 2014. In the design of convolution blocks in GoogleNet, 1×1 convolution with ReLu activation works as a dimension reduction and reconstruct the feature maps [33]; Inception module contains different size of convolution kernels which is helpful to enrich the feature maps.

The inception block was introduced by GoogleNet [3], which uses different kernel sizes. In inception block, 1×1 convolution, 3×3 convolution, 5×5 convolution, and 3×3 Max-pooling are used at the same time using the same convolution. The 1×1 convolution with ReLu activation works as dimension reduction to reconstruct the feature maps [6]. Figure 2.6 shows the inception block in GoogleNet [3].

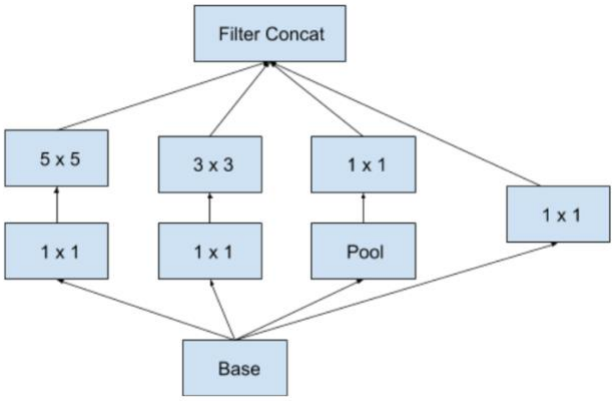


Figure 2.5 Inception module with dimension reduction

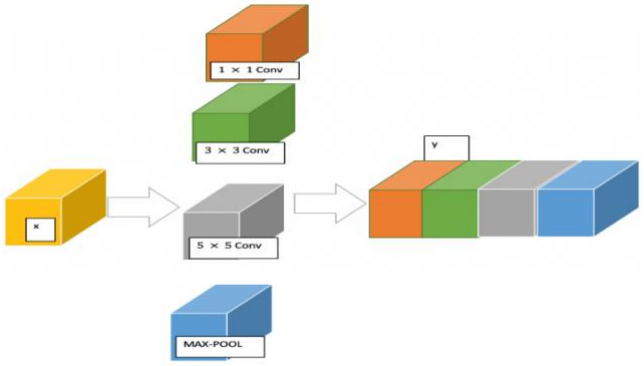


Figure 2.6 Feature Maps for Inception Module.

Inception v2 [31] introduces a concept termed as batch normalization, which is applied to normalizing the value distributions of a layers' output and keep the distribution remain fixed. Inception v3 [23] factorizing convolution is used to reduce parameters. Two kind of factorizing convolutions are introduced, including using small kernel convolutions to replace large convolutions or using asymmetric convolution to replace symmetric convolutions. Figure 2.7 shows a factorization into smaller convolution. The 5×5 convolution area is replaced by two 3×3 convolution areas.

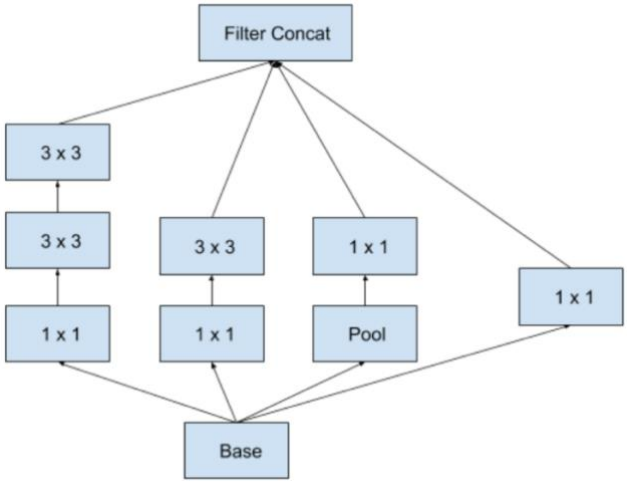


Figure 2.7 Factorization into Smaller Convolution.

Similar with two symmetric 3×3 convolution covering a 5×5 area, asymmetric convolution with one 3×1 followed by one 1×3 convolution can also replace a 3×3 convolution area. The purpose of using the asymmetric convolution is to reduce the number of operation while keep the network's efficiency. With asymmetric convolution, a new version of inception module is shown at Figure 2.8.

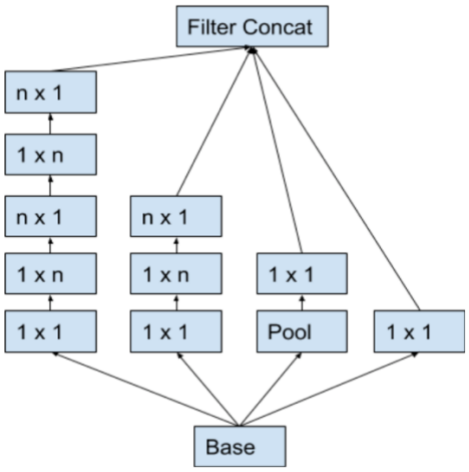


Figure.2.8 Inception Module with Asymmetric Convolution.

Compared with Inception-v3, Inception v4 [32] has more Inception modules. The techniques developed from Inception v1 to Inception v3 are all used to improve model performance. In the Inception-ResNet-v1 and Inception-Resnet-v2, a shortcut connection is added between two activation functions. Three Inception residual block in Inception-ResNet-v1 and Inception Resnet-v2 are shown in Figure.2.9.

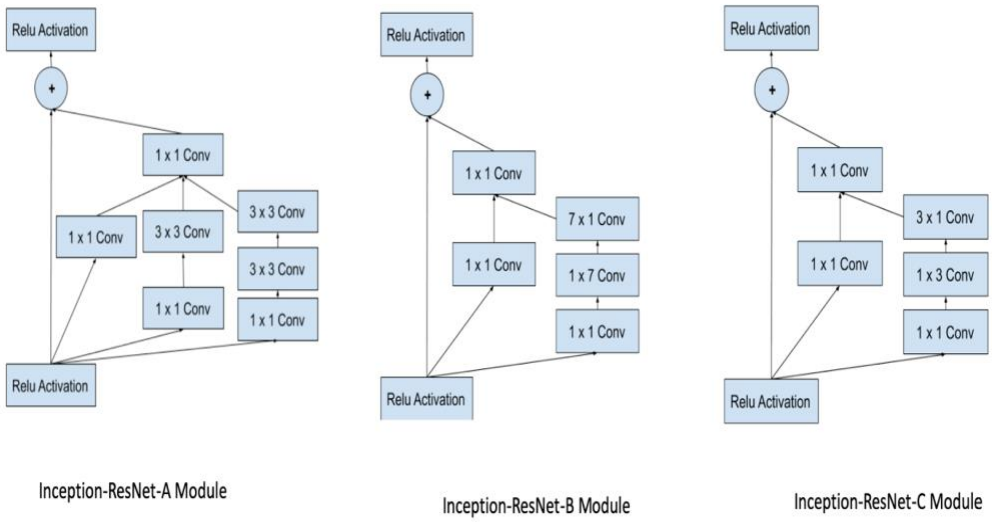


Figure 2.9 Inception–Residual modules in Inception-Residual v2.

2.2 Ecology Datasets

In this classification task, two different ecological datasets respectively are: the bee-wing dataset and the butterfly dataset. The bee-wing is a relatively small and unbalanced dataset and butterfly is a small and relatively balanced datasets. There are 19 classes of New Jersey local bees, which is captured by Dr Gareth Russell's research team, from the biological science department of NJIT. The purpose of this research is to recognize the type of bee only by the image of wings, which is an important part in Dr Russell's research area. The images are captured using a microscope in a 1K by 1K resolution.

There are totally 755 images, including 566 training samples and 189 testing samples. The bee wing dataset contains eight main class in grayscale images, which respectively are agapostemon, augochlora, augochlorella, augochlorella, ceratina, dialictus, halictus and osmia. The first-four type only have one sub-class while the last four type contain more than one sub-class. Ceratina contains three subclasses, which are ceratinacalcarata, ceratinadupla and ceratinametallica. Dialictus contains four subclasses which are dialictusbruneri, dialictusillinoensis, dialictusimitatus and dialictusrohweri. Figure. 2.10 shows sample images for the bee wing dataset and the Figure 2.11 shows the distribution of each class in the bee wing dataset.

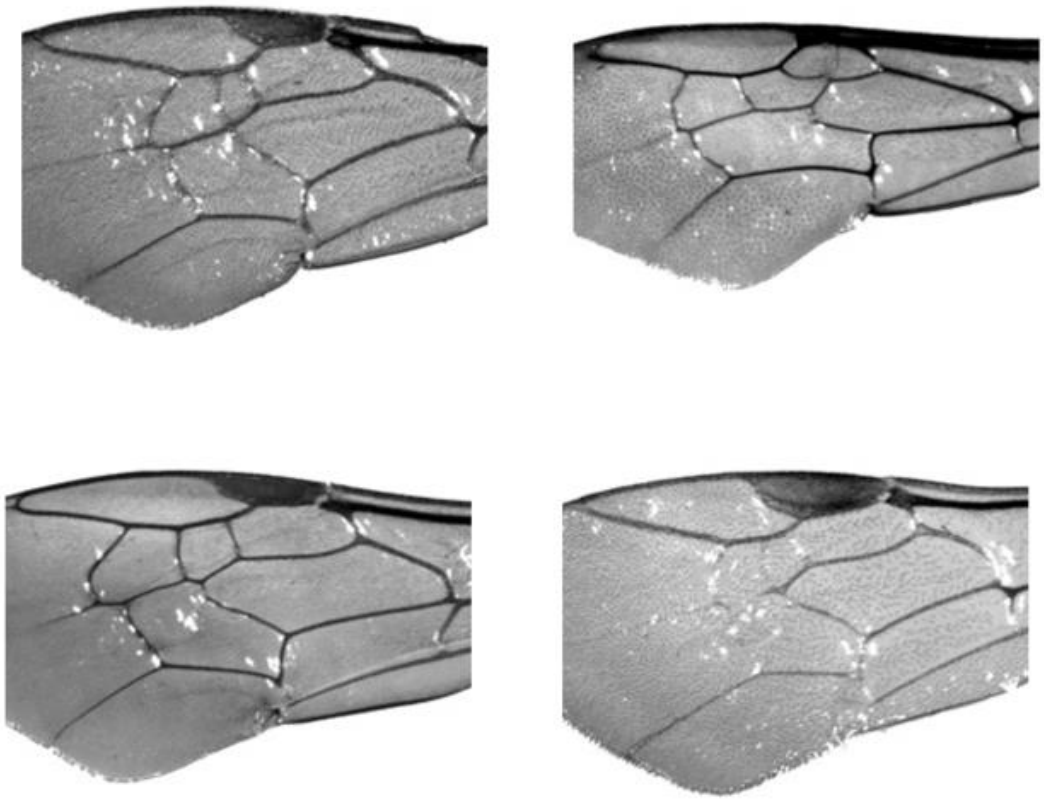


Figure 2.10 Sample image in the Bee Wing Dataset.

.

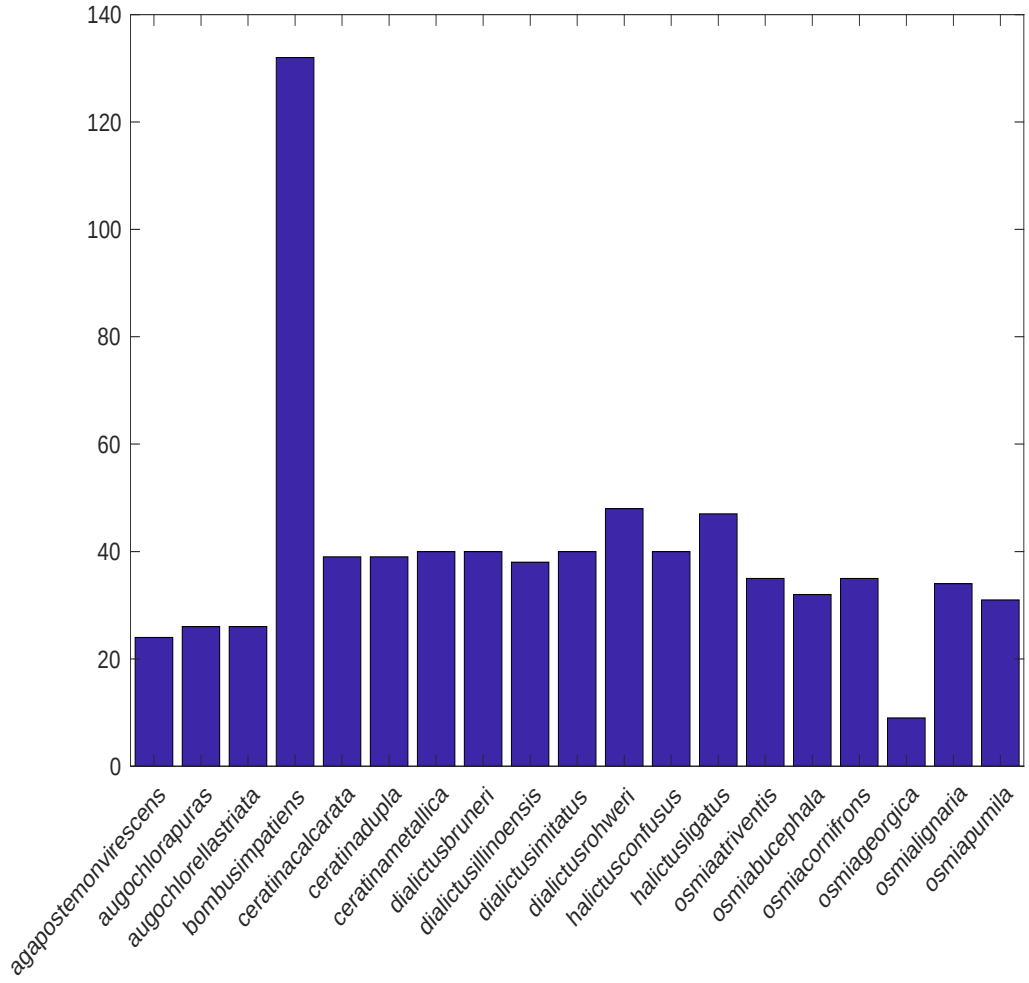


Figure.2.11 Data sample distribution of the Bee Wing Dataset.

The butterfly dataset contains 10 classes of butterfly species, with a range vary from 55 to 100 images per class. The data sample in the butterfly dataset is in RGB format. The total dataset contains 832 image samples, 627 samples for training and 205 image samples for testing. There are ten classes in the butterfly dataset. Figure 2.12 shows data samples and Figure 2.13 shows the data samples' distribution in the butterfly dataset, respectively.

Figure 2.12 Sample image in Butterfly.

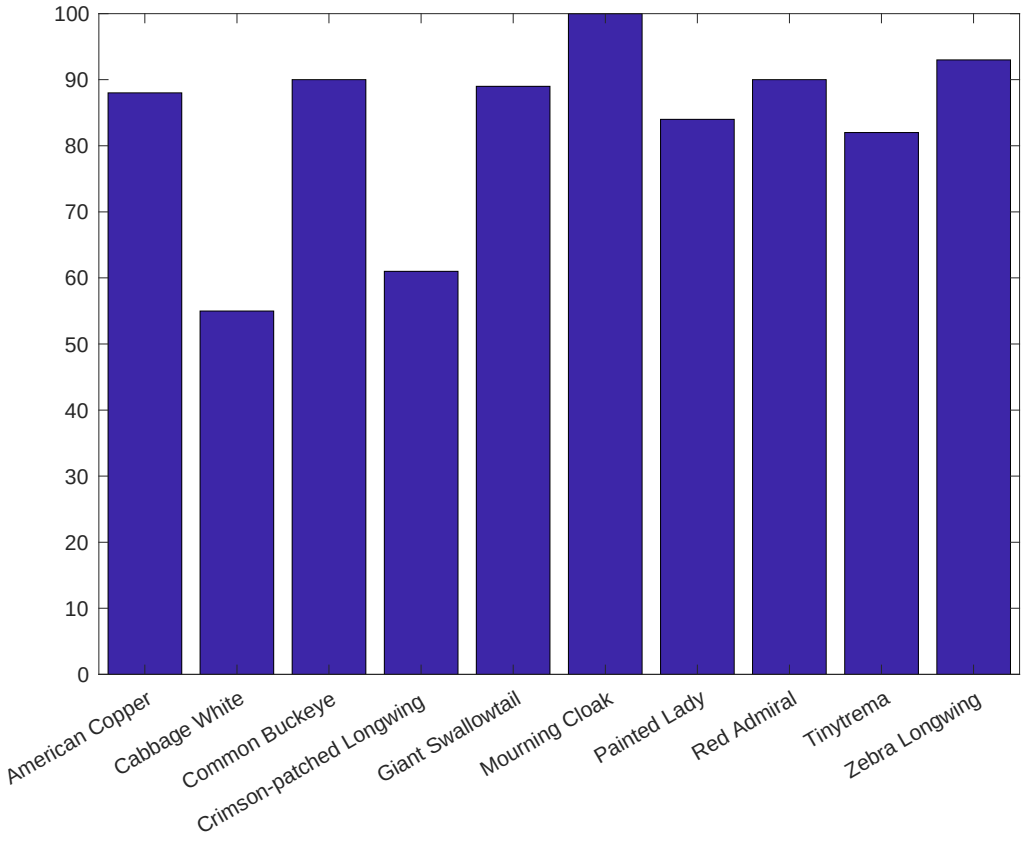


Figure 2.13 Distribution for 10 types of butterfly.

2.3 Classification in Original Dataset

To discovery the best performance for the ecology datasets, seven CNN models, including LeNet-5[1], Alex Net [2], VGG16[4], VGG19[4], Residual Net 50[5], InceptionV3[23], Inception Residue V2[24], are tested with the ecology datasets. The test accuracies are shown in Table 2.1.

Table 2.1: Test Accuracy of the Ecology Datasets

	Bee Wing	Butterfly
LeNet-5	87.78%	70.24%
AlexNet	86.04%	79.85%
VGG16	17.74%	12.17%
VGG19	17.72%	12.28%
ResNet50	86.54%	75.36%
Inception v3	87.16%	78.84%
InceptionResNetV2	87.72%	79.98%

For a small and unbalanced dataset (Bee Wing), a similar test accuracy is achieved at nearly 87%, except for VGG16 & VGG19. Considering LeNet is a two-layer convolutional neural network and a similar test accuracy is achieved in Inception-V3 and Inception-ResNet-V2, the feature in this dataset is a relatively simpler than the butterfly dataset and can be extracted by a two-layer CNN. The feature in bee wing dataset is mainly lines or blobs also indicate the CNN models do not need to extract this feature from a much more complicate background.

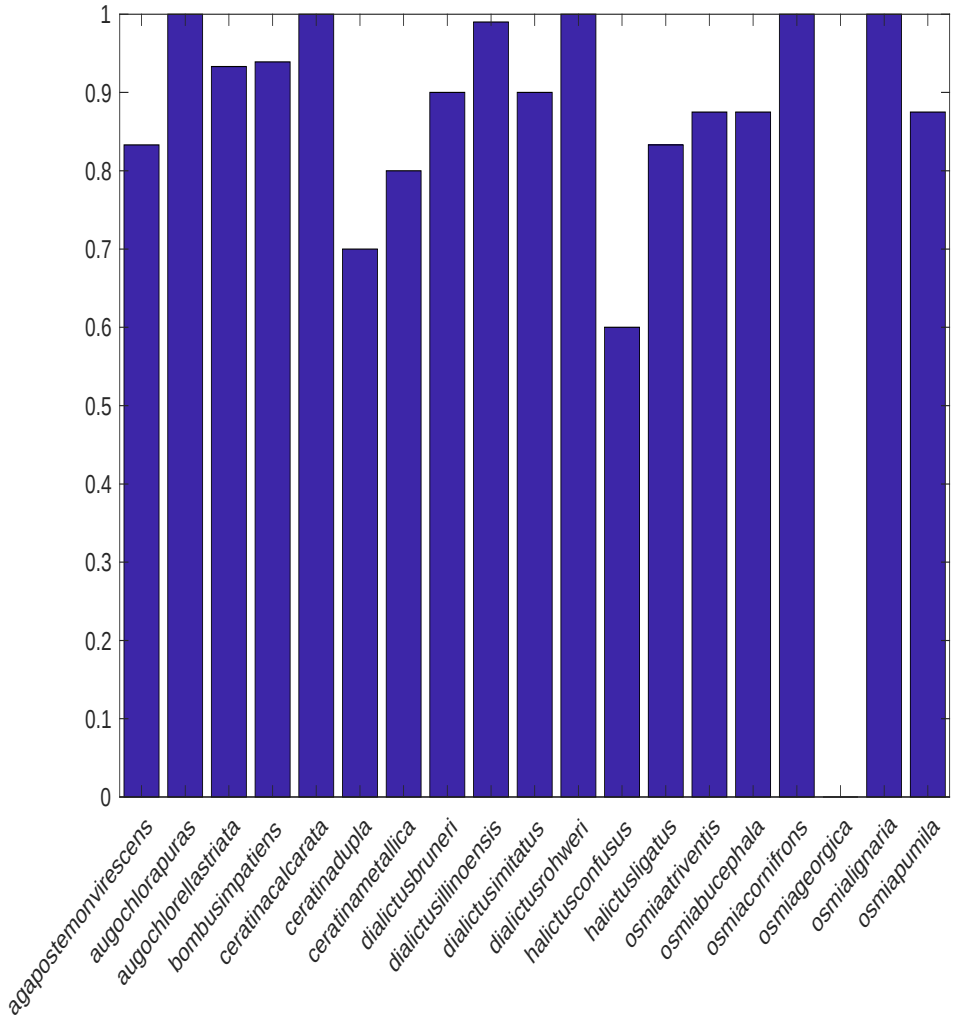
VGG 16 and VGG 19 model are facing a convergence problem in training, it is probably due to limited data caused underfitting or a vanishing gradient problem. Researches in [3] [4] shows that with the increasing of complexity of a CNN model, a deeper neural network may have a high possibility to have difficulties in convergence. However, the problem in VGG-Net did not show in Reset50. This is due to Residual Neural Network uses residual connections to avoid vanishing gradient problem.

Inception v3 uses an inception blocks with different convolution kernel size to enrich the feature maps; Inception Residual Neural network combine inception blocks with residual connection. With a residual block, Inception v2 model achieves a higher test accuracy than Inception v3 model.

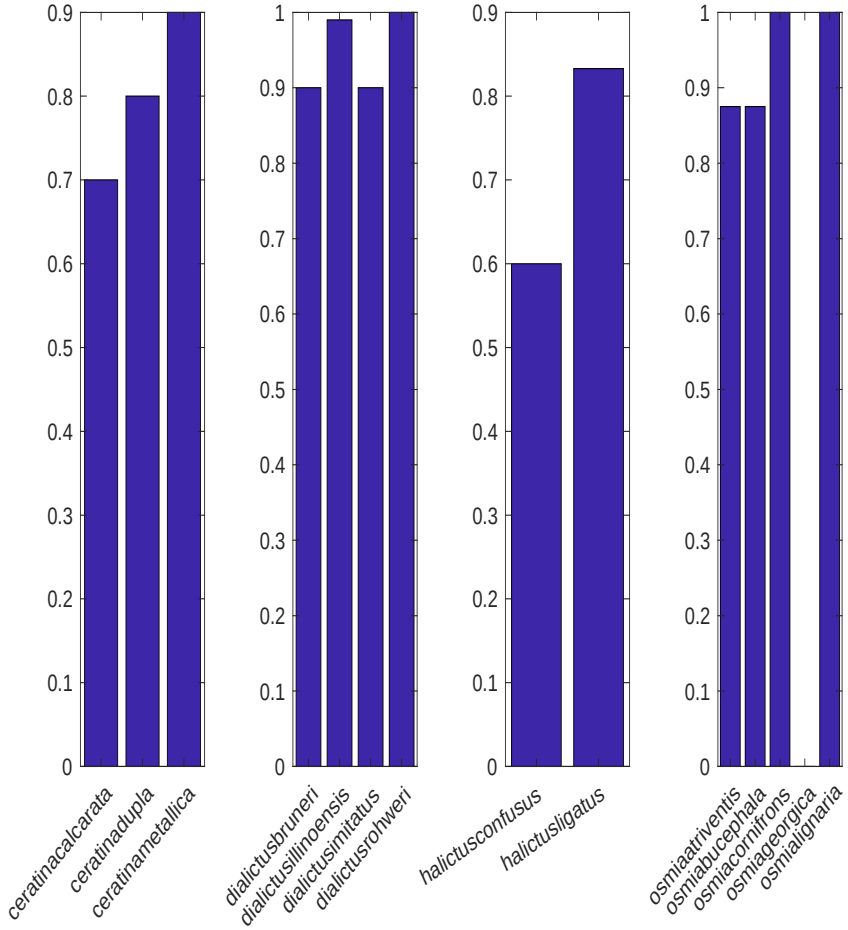
Also, the low-test accuracy in bee-wing is due to the effect form sub-species which may have more common features. The single class test accuracy of each dataset is shown in Figure 2.14. A relatively lower test accuracy is achieved between sub-class species. In ceratina class, ceratinadupla's single class achieved a test accuracy of 70%, 17% lower than the overall accuracy. And in halictus, halictusconfusus achieved a test accuracy of 60%, 27% lower than the overall accuracy. In osmia, osmiageorgica achieved a test accuracy of 0, both of the two samples are classified to osmiageorgica, another sub-class in osmia. Figure 2.14(c) shows a heap map of the confusion matrix.

Although given the fact that subclass species are closely to each other and an insufficient data sample obstruct feature learning process, a class of bee-wing achieved 0 performance should be aware. This phenomenon signifies a close impossibility for this classifier to recognize any it's related target. It also attracts ecologists' attention especially when they are trying to build a specie classifier or ecology ID system.

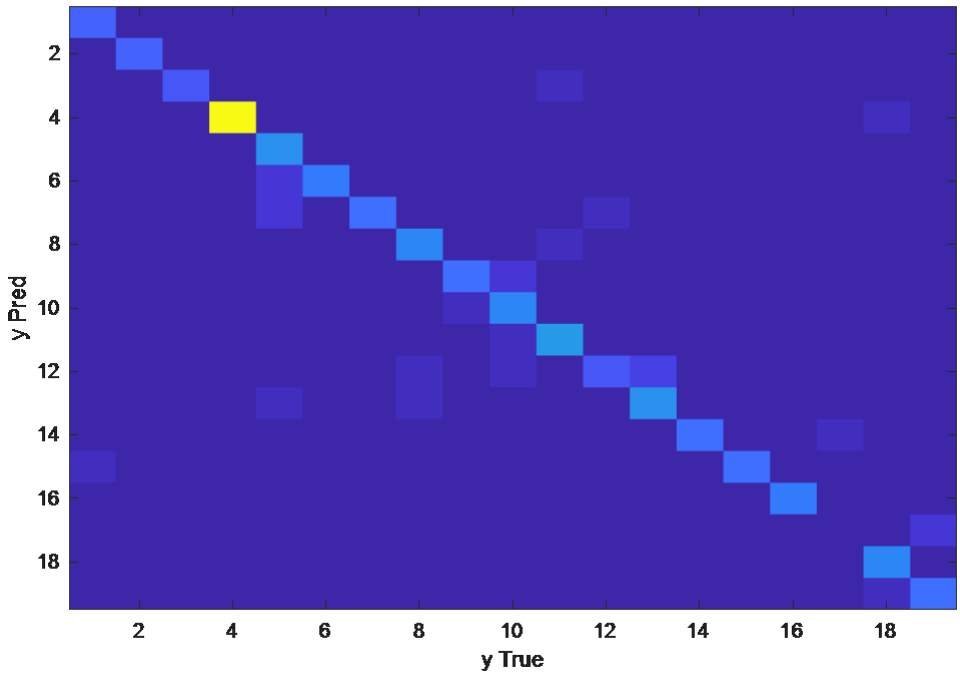
Ecologists are focusing on increase the possibility to recognize the minority class of species and improve model performance. Future work will be focused on increasing the model's ability to recognize specie with little data samples.



(a) Each class accuracy: Bee Wing



(b) Wing (b) Bee-Wing subclass classification



(c) Heatmap of confusion matrix (labels from 1-19, represent from agapostemonvirescens to osmiapumila)

Figure 2.14. Each class classification rate and Bee-wing subclass classification rate.

For a small and relatively balanced dataset(butterfly), two similar test accuracies close to 79% are achieved in AlexNet model and InceptionResV2 model. The reason that LeNet achieve a low accuracy at 70% is partially due to this dataset contains complexed background and need more convolution layers to extract features from background.

VGG16 and VGG19 models are facing a similar convergence problem in this bee-wing dataset. A 75% test accuracy is achieved in ResNet50 shows residual connection is helpful for models to go deeper. The low-test accuracy also due to an

insufficient dataset. InceptionRes v2 models are achieved a higher test accuracy than Inception v3, shows a promising feature extraction ability for inception residual block.

In order to solve the low-test accuracy problem for small datasets, two approaches in deep learning are applied to make improvement in Bee-wing and butterfly, respectively are data augmentation and transfer learning.

2.4 Data Augmentation

Data augmentation is a technique that artificially generate new images from the original dataset. Compared to the large dataset samples usually used in training a CNN model, the original data in bee wings dataset and butterfly are relatively small. By using data augmentation technique, the amount of data samples can be enlarged based on original dataset while at the same time keeps the features from original dataset. Thus, the first approach to improve model's performance is by using data augmentation techniques to enlarge the dataset. Data augmentation is by performing a sequence of image-processing operations to the original image. This operations including perspective skewing, elastic distortion, rotation, mirroring and cropping. The following operations focus on changing the images from different view angles and does not change the features in these images.

The tool to create an augmented dataset is called Augmentor [26]. The process of creating an augmented dataset is as follow. First, image-processing functions are performed sequentially through a pipeline. Then, a set of predefined probability is applied to control the probability of each image processing operation. After that, a large

number of new images depending on the number of operations and the range of values used in each operation.

Perspective skewing is referred to an image transformation whose effect is viewing this object from different angles. Users can define a direction to perform skewing. Figure 2.15 shows the augmented images from bee wing dataset after perspective skewing functions are applied. Figure 2.16 shows the augmented images from butterfly dataset after perspective skewing functions are applied.

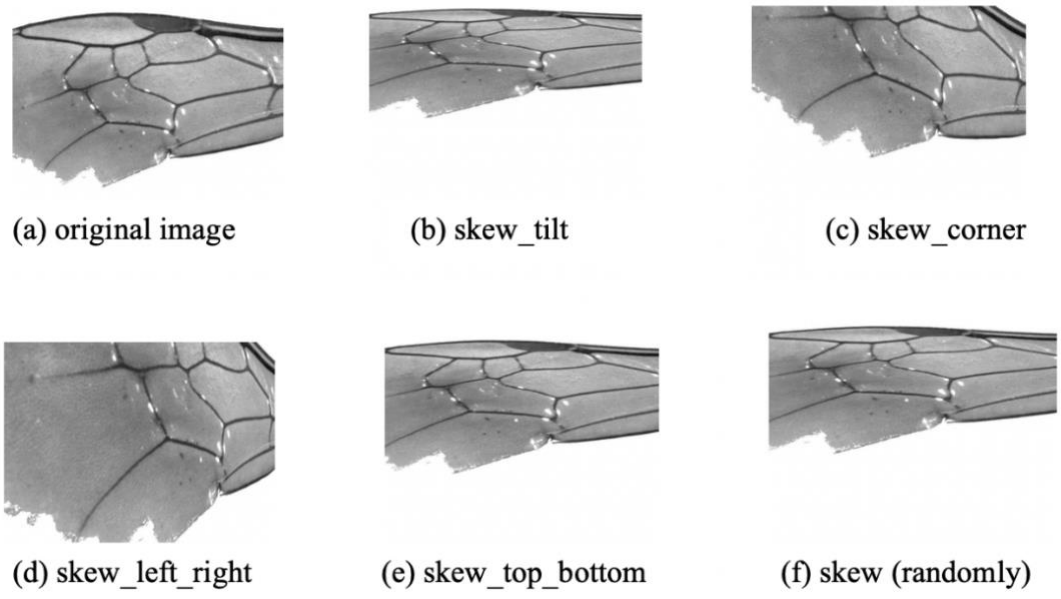


Figure 2.15 Perspective skewing performed on the Bee Wing Dataset. (a) Original image, (b)-(e) the images after performing perspective skewing to a certain direction, (f) the image after performing perspective skewing to a random direction.

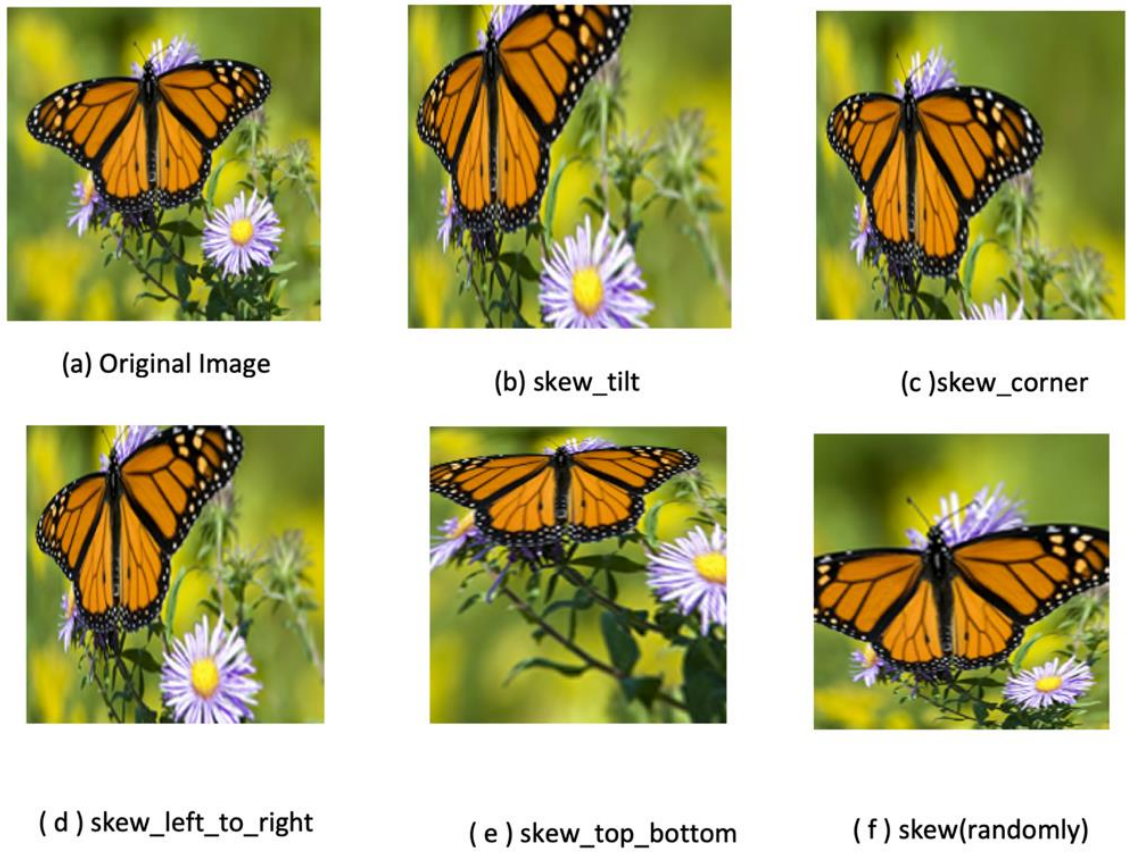


Figure. 2.16 Perspective skewing performed on the Butterfly Dataset. (a) Original image, (b)-(e) the images after performing perspective skewing to a certain direction, (f) the image after performing perspective skewing to a random direction.

Elastic distortion is a function that allows users to make random distortions on the original image, while the image's aspect ratio is still maintained. Figure 2.15 shows the augmented images from bee wing dataset after elastic distortion functions are applied; Figure 2.16 shows the augmented images from butterfly dataset after elastic distortion functions are applied.

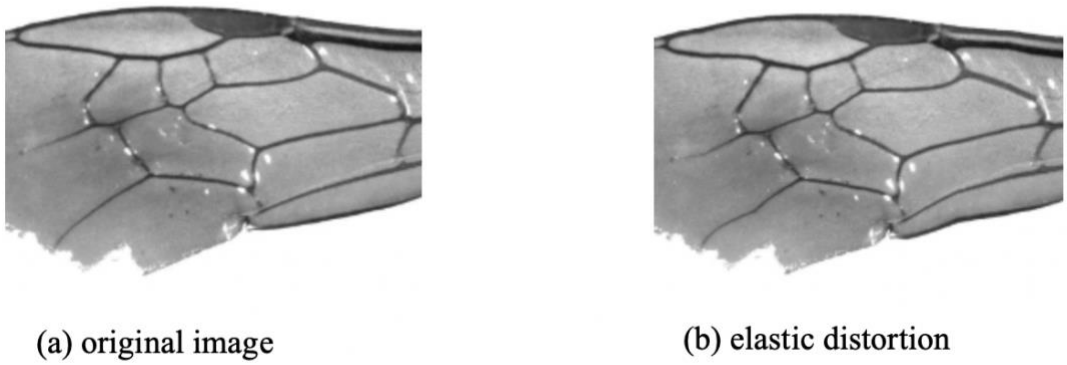


Figure. 2.17 Elastic Distortion on the Bee Wing Dataset. (a) Original image and (b) the image after elastic distortion.

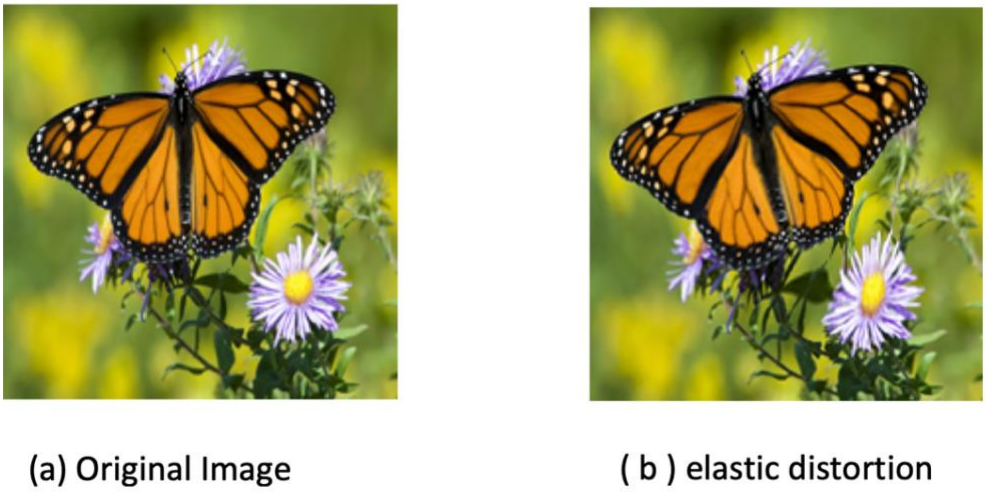


Figure. 2.18 Elastic distortion on the Butterfly Dataset. (a) Original image and (b) the image after elastic distortion.

Rotation is a function to rotate an image in a number of ways, such as rotating 90°, 180°, or 270°. However, it could be performed by a random degree, which incorporates zoom-in or zoom-out from the original image. Figure 2.19 shows the

augmented images from bee wing dataset after rotation functions are applied; Fig 2.20 shows the augmented images from butterfly dataset after rotation functions are applied.

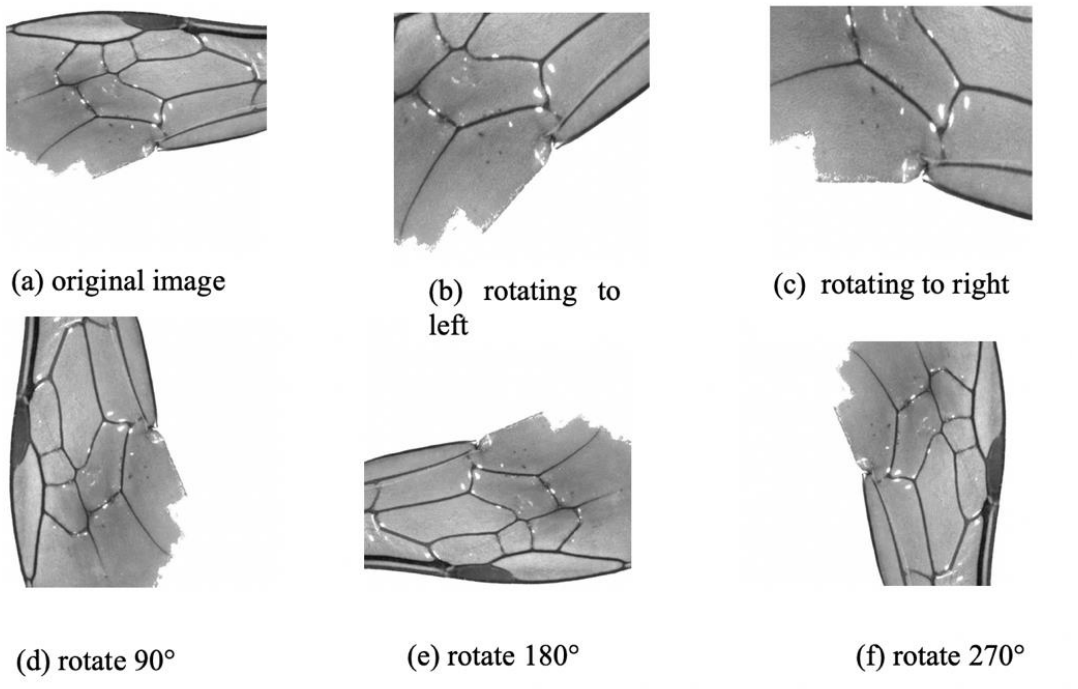


Figure 2.19 Rotation on the Bee Wings Dataset. (a) Original image, (b) and (c) rotated by two random angles (range is set from -45° to 45°) with a zoom-in effect, (d)-(e) rotated by 90°, 180°, or 270°, respectively.

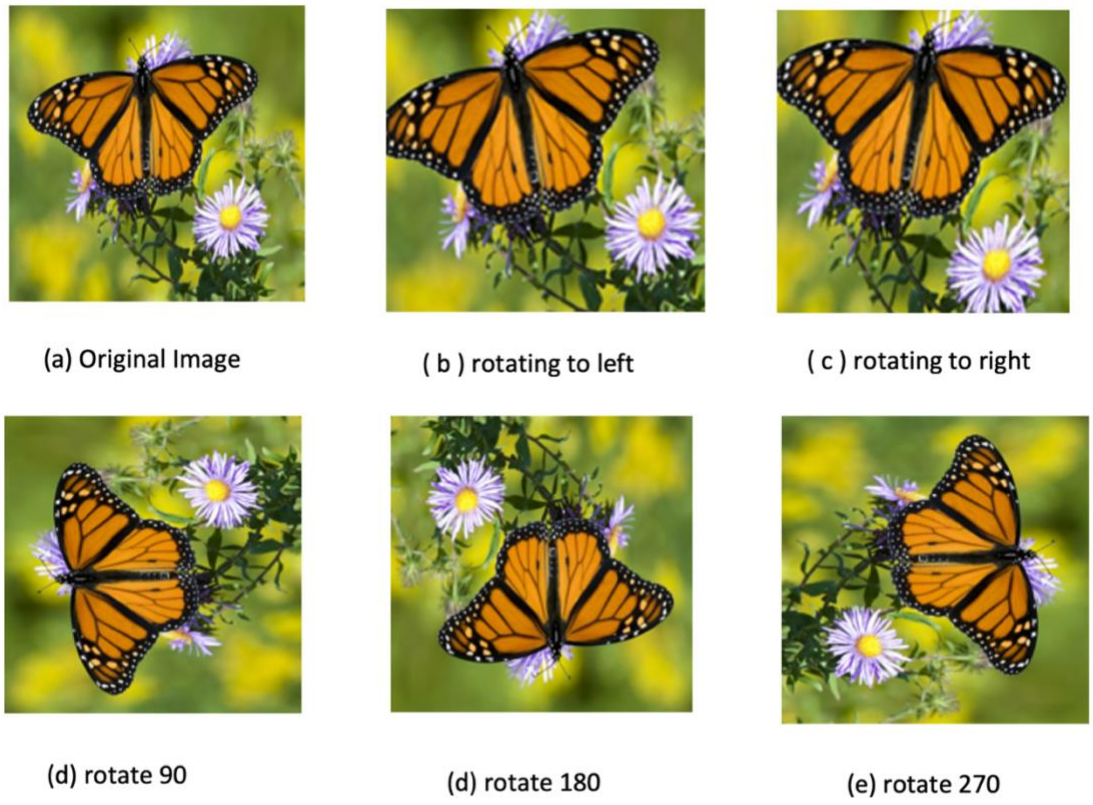


Figure 2.20 Rotation on the Butterfly Dataset. (a) Original image, (b) and (c) rotated by two random angles (range is set from -45° to 45°) with a zoom-in effect, (d)-(e) rotated by 90°, 180°, or 270°, respectively.

Shearing is a function that tilts an image along one of its sides. It can be tilted from left-to-right or right-to-left. Fig 2.21 shows the augmented images from bee wing dataset after shearing functions are applied; Fig 2.22 shows the augmented images from butterfly dataset after shearing functions are applied.

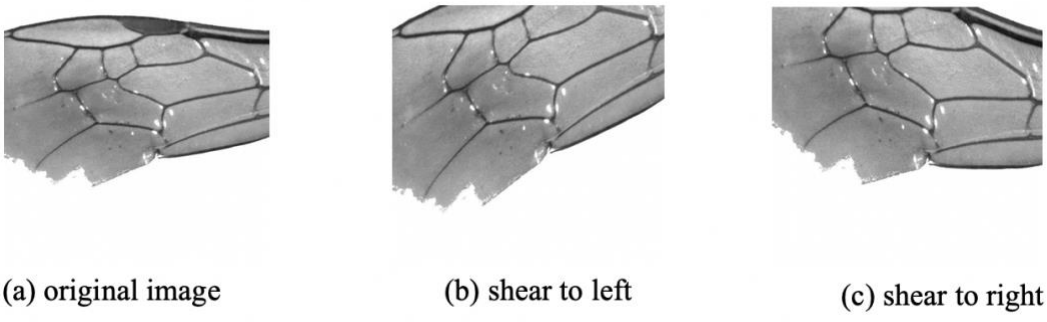


Figure 2.21 Shearing on the Bee Wing dataset. (a) Original image and (b) shearing to random directions

Figure 2.22 Shearing on the Butterfly Dataset. (a) Original image and (b) shearing to random directions

Mirroring is a function that reflect duplication of an object that appears almost identical but is reversed in the direction perpendicular to the mirror surface. Figure 2.23 shows the augmented images from bee wing dataset after mirroring functions are applied; Figure 2.24 shows the augmented images from butterfly dataset after mirroring functions are applied.

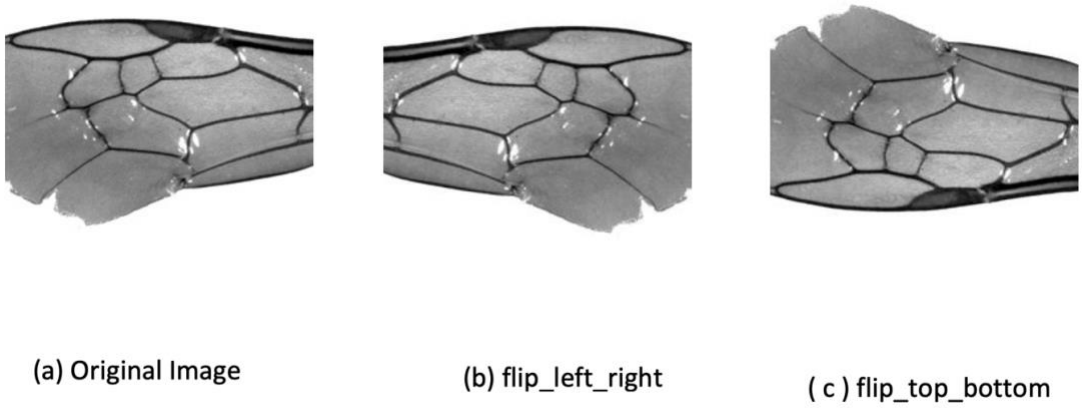


Figure 2.23 Mirroring on the Bee Wing Dataset. (a) Original image (b) flip_left_right (c)flip_top_bottom

Figure 2.24 Mirroring on the Butterfly Dataset. (a) Original image (b) flip_left_right (c)flip_top_bottom

Cropping is the removal of unwanted outer areas from a photographic or illustrated image. Figure 2.25 shows the augmented images from bee wing dataset after

cropping functions are applied; Figure 2.26 shows the augmented images from butterfly dataset after cropping functions are applied.

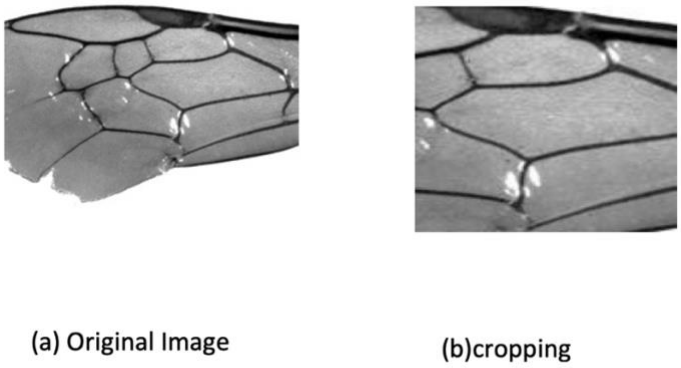


Figure.2.25 Cropping on the Bee Wing Dataset. (a) Original image (b) cropped image

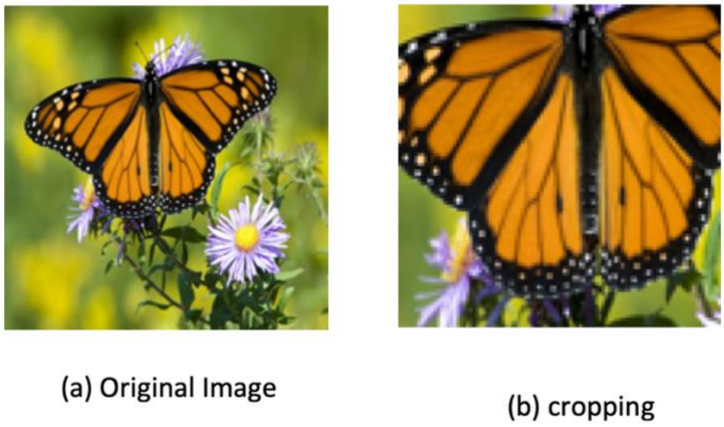


Figure.2.26 Cropping on the Butterfly Dataset. (a) Original image (b) cropped image

2.5 Transfer Learning

Transfer learning is referred as a machine learning concept that gains knowledge from one task and reuse it to fulfill a different task [28]. In deep learning, transfer learning is often conducted by using a well-trained model which previously been trained on a large dataset and then utilize the parameters for another task. SInce The ecology dataset does not have a sufficient size to train an entire CNN with random initialization. So pretrain deep learning model on a large dataset and train from scratch is an approach to solve this problem. Several pre-trained models that have been trained on ImageNet [29] are used for transfer learning model. These models including VGG16, VGG19, ResNet50, InceptionV3, InceptionResV2.

According to [30], in a deep convolution neural network, some features are learned from convolutional neural networks that contain more common features, such as edge detectors or color blob detectors, which can be used in many other tasks. The later layers become progressively more specific to the details of the classes contained in the original dataset. The design for using transfer learning takes the following steps: First, using a pre-trained CNN model which been trained on ImageNet and replace the previous fully connected layers. Second, add new fully-connected layers and use the model to train for ecology datasets. At last, fine-tune some higher-level portion of the network.

2.6 Re-designed Convolution Blocks

In the inception models, different convolutional kernel sizes are used for feature extraction. Inspired by this idea, we redesign the inception block and the inception residual block using four convolutional kernels, which are 1×1 Same Conv, 3×3 Same Conv, 5×5 same Conv, and 7×7 same Conv. The outputs are concatenated together and then passed to a 1×1 Conv. We replace the max-pooling layers by 7 x 7 same convolution to include a larger convolution kernel for detecting a wider and larger area. By combining more information in feature map, the CNN model can be more sensitive in telling the difference among different classes.

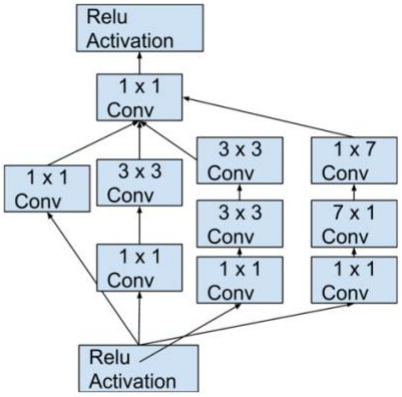


Figure 2.27 Re-designed Inception block.

The inception residual block contains four different size of convolution kernels, which are 1×1 Conv, 3×3 Conv, 5×5 Conv, 7×7 Conv and a residual connection from block input to block output. The residual may help if the weight in inception block is not well trained. Figure 2.29 shows the Inception Residual blocks.

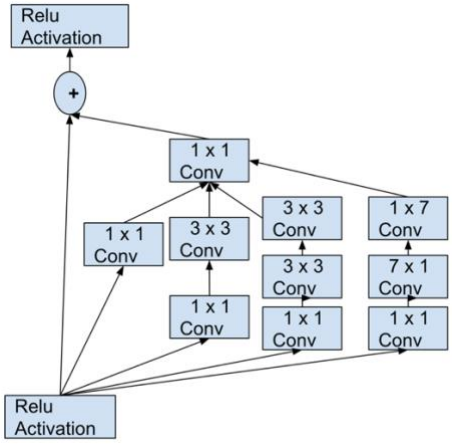


Figure 2.28 Re-designed Inception Residual Block

By using a different number of convolution blocks and subsampling layers in the bee wing dataset, we can compare the performance of redesigned inception block and inception residual block. shows the model to compare the redesigned inception and the inception residual block.

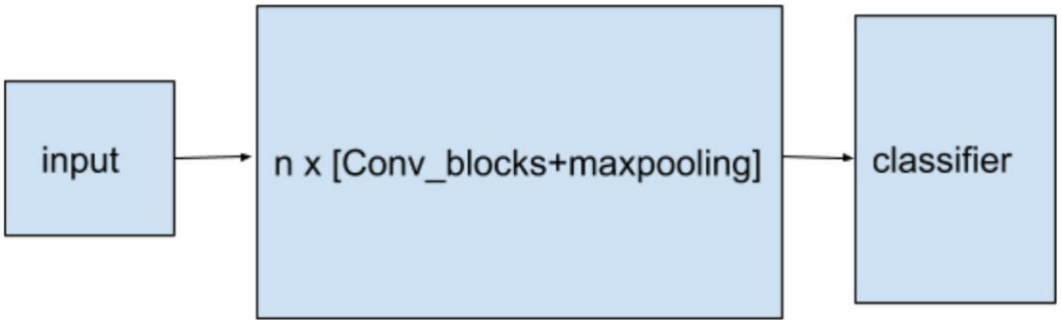


Figure 2.29 Different $n \times$ Conv_blocks classification model

2.7 Experimental Results

Table 2.1 Test Accuracy of the Ecology Datasets

	Bee Wing	Butterfly
LeNet-5	87.78%	70.24%
AlexNet	86.04%	79.85%
VGG16	17.74%	12.17%
VGG19	17.72%	12.28%
ResNet50	86.54%	75.36%
Inception v3	87.16%	78.84%
InceptionResNetV2	87.72%	79.98%

The test accuracy in original dataset is shown in Table 2.1. Bee wing achieve a test accuracy among 86% ~ 87% in LeNet, AlexNet and Inception models. Butterfly achieve a similar test accuracy among 78%~79% in AlexNet and Inception models. To improve the performance for bee wing and butterfly, data augmentation, transfer learning, and data augmentation with transfer learning are applied.

Table 2.2 Test accuracy for bee wing dataset

Bee Wing	Original dataset	Data Augment	Transfer Learning	Transfer & Aug
LeNet-5	87.78%	89.97%	_	_
AlexNet	86.04%	89.8%	90.37%	91.28%
VGG16	17.74%	88.7%	92.58%	93.41%
VGG19	17.72%	87.34%	91.67%	93.19%
ResNet50	86.54%	89.34%	92.5%	93.12%
Inception v3	87.16%	91.46%	92.28%	93.95%
InceptionResNetV2	87.72%	90.91%	92.97%	94.40%

Table 2.2 shows the test accuracy of the bee wing dataset. The test accuracy in original dataset shows a similarity test accuracy at 87%. By applying data augmentation, the test accuracy gets improved in each model. A similar test accuracy close to 90% is shown by using LeNet, AlexNet and Inception models. Also, data augmentation helps to improve VGG 16 and VGG19 models' convergence problem in training with limited samples of data.

Transfer learning also improved the test accuracy with the original dataset. VGG19 shows the best test accuracy at 94.67% and inception models shows a common performance at 90%, indicate a well-trained VGG19 model do not need to select a suitable kernel size and has an ability to achieve a better performance in bee wing dataset.

By combine the data augmentation and transfer learning, a similar test accuracy at 94% is achieved. These improvements prove the effectiveness of using data augmentation, transfer learning and their combination in small dataset classification problems.

Table 2.3. Test Accuracy of the Butterfly Dataset

Butterfly	Original dataset	Data Augment	Transfer Learning	Transfer & Aug
LeNet-5	70.24%	71.41%	_	_
AlexNet	79.85%	80.83%	89.28%	92.75%
VGG16	17.74%	79.91%	90.65%	95.04%
VGG19	17.72%	80.33%	90.73%	94.66%
ResNet50	79.21%	86.54%	92.60%	96.88%
Inception v3	80.32%	87.16%	93.10%	96.10%
InceptionResNetV2	81.94%	87.72%	93.67%	96.07%

Table 2.3. shows the test accuracy for butterfly dataset. In original dataset, LeNet achieves a 70.24% test accuracy and AlexNet shows a test accuracy at 79.85% proves a deeper convolution models can improve the models' performance. By using data augmentation, a slightly improvement is made for each model. This may indicate the data augmentation failed to improve the diversity of this small dataset by only performing image transformations. But transfer learning provides more generated information from a pre-trained model. By combining the data augmentation and transfer learning, the performance improved much better than bee wing dataset. The

test result in butterfly dataset also improved the effectiveness of transfer learning and data augmentation.

The test result with original dataset for using different number of inception and inception residual block is shown at Table 2.4 and the test result with augmented dataset for using different number of inception and inception residual block is shown at Table 2.5.

Table 2.4. Test accuracy for inception and inception residual models (Original dataset)

Original Dataset	Inception Block	Inception residual Block
2 × Blocks	90.04%	92.89%
3 × Blocks	90.04%	92.05%
4 × Blocks	89.24%	92.09%
5 × Blocks	88.75%	92.90%

Table 2.5. Test accuracy for inception and inception residual models (augmented dataset)

Augmented Dataset	Inception Block	Inception residual Block
2 × Blocks	90.31%	93.05%
3 × Blocks	90.96%	92.44%
4 × Blocks	89.93%	92.34%
5 × Blocks	89.90%	92.40%

In Table 2.4, different number of Inception blocks and Inception residual blocks are used in original bee wing dataset. The test accuracy for 2x inception block is 90.04% and for 2 x inception residual block is is 92.89%, while LeNet achieves an accuracy of 87.78%. In Table 5, different number of Inception blocks and Inception residual blocks are used in augmented bee wing dataset. The test accuracy for 2x inception block is 90.31% and for 2 x inception residual block is 93.05%, while LeNet achieves an accuracy of 89.87%.

Compared with inception block, Inception residual block achieves a better test accuracy. The experiment result proves the Inception residual block has the ability to achieve a higher performance in feature extraction.

2.8 Summary

First, different deep learning models are used to train the ecology datasets. Due to a small sample dataset problem, the test accuracy for bee wing is achieved at 87% and for butterfly is achieved at 79% except for VGG16 and VGG19 models. VGG16 and VGG19 also shows a poor ability in training for a small sample dataset with deeper convolutional layers. Because a small data sample problem causes model underfitting and a stacked convolution connection cause vanishing gradient.

To solve the following problem in original dataset, data augmentation and transfer learning are used to improve the performance of the deep neural network. The experiment result shows data augmentation improves the test accuracy slightly may suggest that by only using image transformation technique cannot provide enough feature for the learning models. Transfer learning can help to improve the test accuracy in small datasets by first learning from a large dataset and fine-tuned in the original ecology dataset. Also, the combination of these two methods can help to improve to a higher test accuracy of 94% for bee wing and 98% to butterfly by providing the pre-trained model with more data samples. Also, by using data augmentation technique, the VGG16 and VGG19 models conquer the problem of underfitting. And by using transfer learning, a pre-trained VGG16 or VGG19 model conquered the problem of vanishing gradient in small dataset.

Finally, a comparison between using inception block and inception residual block in bee wing dataset suggest the redesigned inception residual block has an advantage in maintaining its advantage when model goes deeper.

Chapter 3

CLASSIFICATION OF ECOLOGY IMAGES USING MORPHOLOGICAL NEURAL NETWORK

Deep learning [38] is an essential part in machine learning, which requires a large amount data to train a model and then evaluate the model's performance on different datasets. In this section, we present the basic structure of convolution neural networks, the mathematical morphological operations, and the morphological neural networks.

3.1 Morphological Neural Network

3.1.1 Mathematical Morphological Operations

In computer vision, the convolutional neural networks are widely used in many areas. The basic deep learning framework contains an input layer, a feature extraction layers, and a pooling layer to reduce unnecessary data. After the feature extraction layers, the feature representations are fed to a fully connected artificial neural networks for classification. Typically, the input is one or several images with one or three channels, which could be grayscale or RGB images. Traditional CNN models perform convolution operations for several times with different filters, so there are several output images, called feature maps. In this part, a different and novel feature extraction mechanism, the Mathematical morphology, instead of convolution, is presented and shows its effectiveness.

Mathematical morphology is a widely used approach for shape representation and image preprocessing. Two fundamental morphological operations are dilation and

erosion. Let the input image be I and the structuring element be s. The dilation operation is denoted as $I \oplus s$, which expands the image by the structuring element. The erosion is denoted as $I \ominus s$, which shrinks the image by the structuring element. Other often used morphological operations are opening, closing.

The opening is typically used for contour smoothing, especially for breaking thin connections between components and enlarging small holes or gaps. It is defined as an erosion followed by a dilation as the equation (3.1).

$$I \circ s = (I \ominus s) \oplus s \tag{3.1}$$

Different from opening, the closing can be used for connecting narrow areas and filling in small holes or gaps. It is defined as a dilation followed by an erosion as as the equation (3.1).

$$I \bullet s = (I \oplus s) \ominus s \tag{3.2}$$

Figure 3.1 shows two sample images for chest X-ray images, which are processed using dilation and erosion with a 6×6 structure element of all 1's. Figure 3.2 shows two sample images, which are processed using closing and opening with a 6×6 structure element of all 1's.

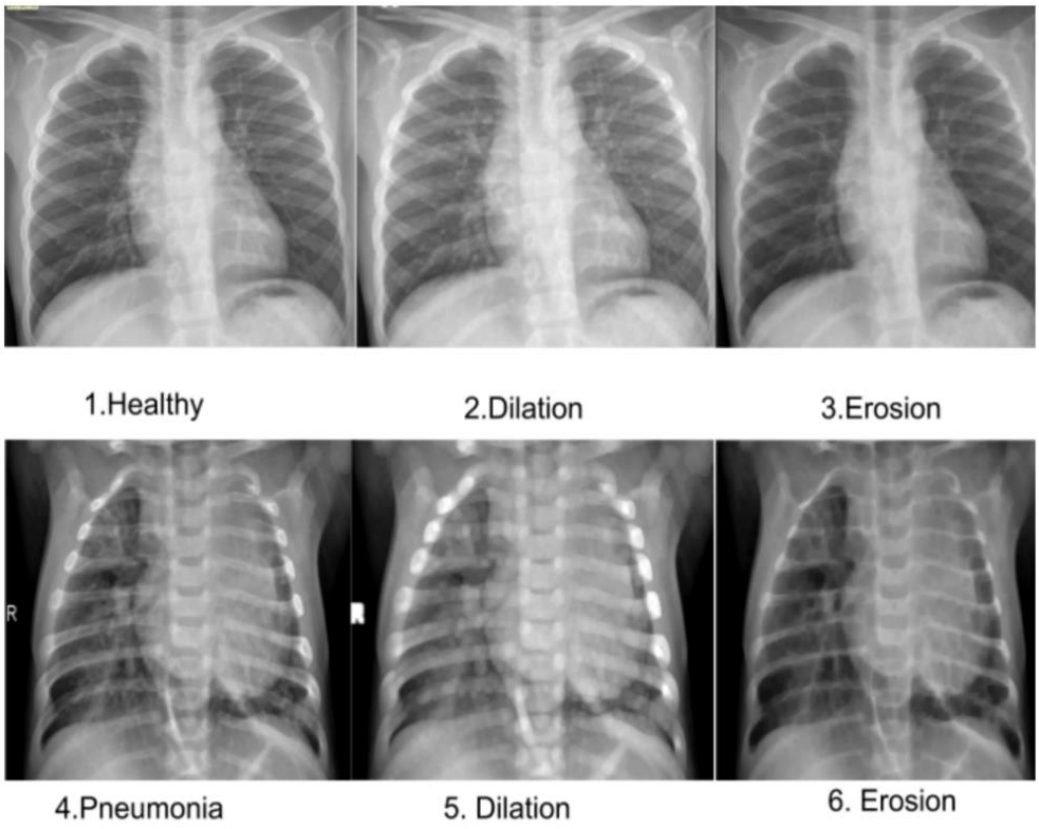


Figure 3.1 Sample images after morphological operations. Column 1 shows input images; column 2 shows dilation; column 3 shows erosion.

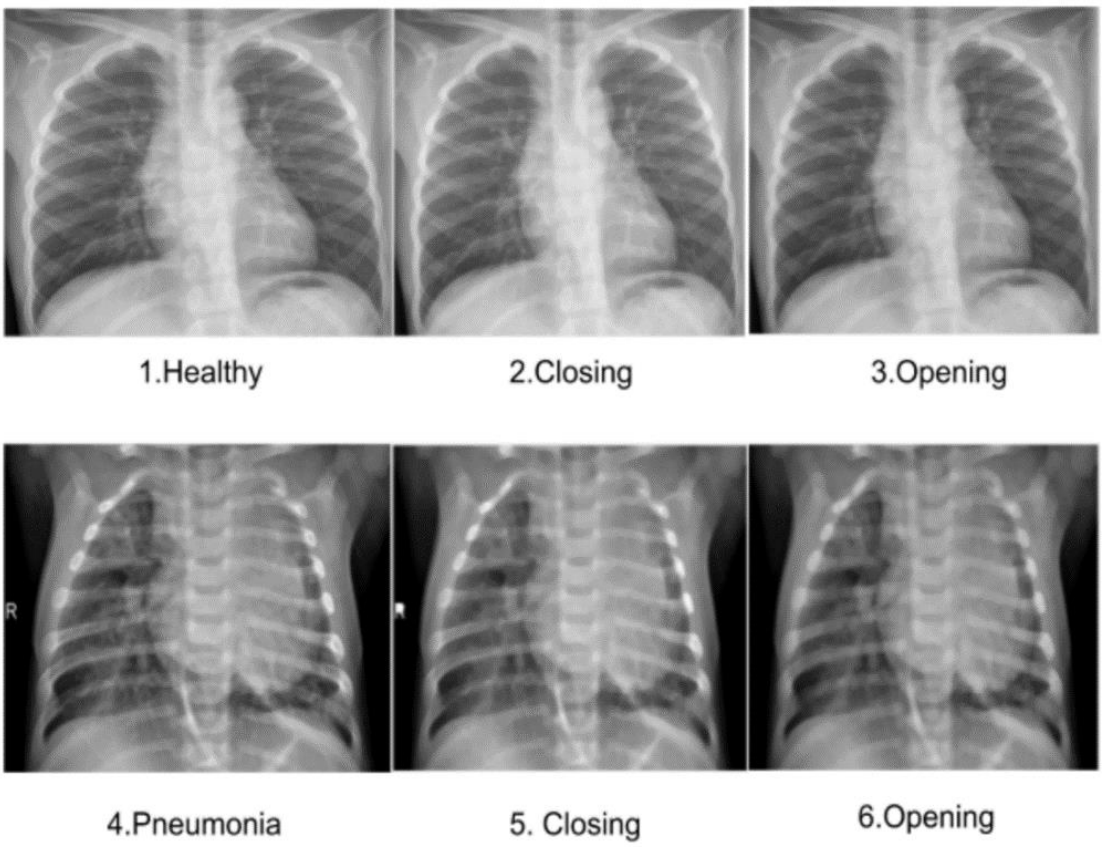


Figure 3.2 Sample images after morphological operations. Column 1 shows input images; column 2 shows closing; column 3 shows opening.

For the X-ray images, the dilation operation can expand some of the small areas while enlarging some of the noisy areas. The erosion can clean the background by eliminating some noisy areas, but at the same time, filtering out some pixels. Opening

and closing can smooth the contour, where closing tends to fill in some holes and opening tends to make them larger. Other usually used morphological operations including the top-hat transformation operation and the bottom-hat transformation. The top-hat transformation is denoted as $I - I \circ s$, and the bottom hat transformation is denoted as $I \bullet s - I$.

3.1.2 Morphological Layers

The morphological neural network (MNN) is another type of deep learning framework. Similar to the convolutional layers in CNN, the morphological layers work as a feature extraction tool. Shih et al. [5] proposed the development of deep learning framework for two morphological layers: the dilation layer and the erosion layer. For the j-th pixel in an output image Y, the dilation layer is defined as equation (3.3)

$$Y_j = \ln(\sum_{i=1}^{n} e^{W_i X_i}) \tag{3.3}$$

W represents the corresponding structure element and X represents the input image. For the j-th pixel in an output image Y, the erosion layer is defined as equation (3.4):

$$Y_j = -\ln(\sum_{i=1}^{n} e^{-W_i X_i}) \tag{3.4}$$

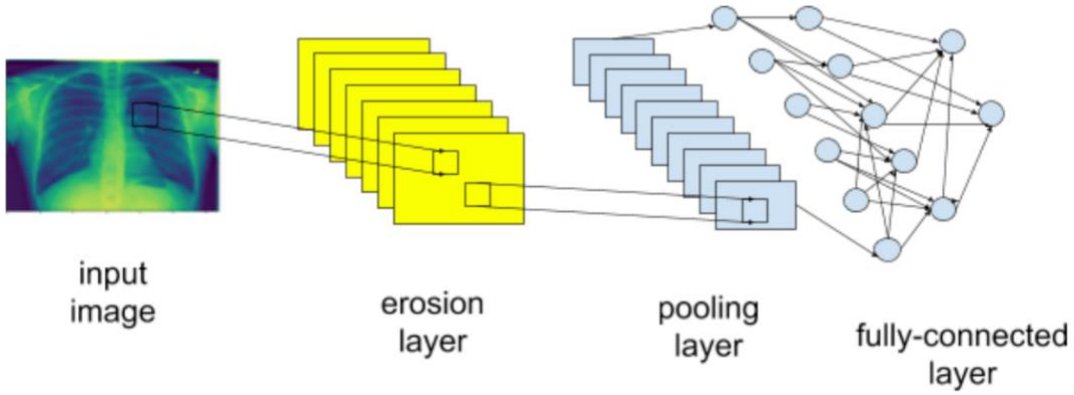
3.2 Basic Morphological Neural Network Design

In this section, we present different deep learning models for the classification of ecology images. Different mathematical morphological operations, such as dilation, erosion,

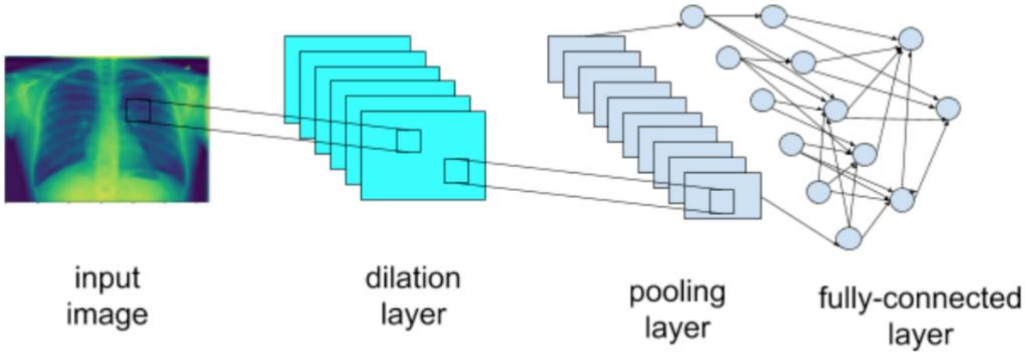
closing, opening top-hat and bottom-hat, are developed with different combinations of morphological layers. These models require to specify the operation types before training the deep neural networks. To solve this problem, morphological neural networks using adaptive layers are proposed and applied for pneumonia classification. These models do not require to specify the morphological operation types for each layer.

3.2.1 Basic Morphological Neural Networks

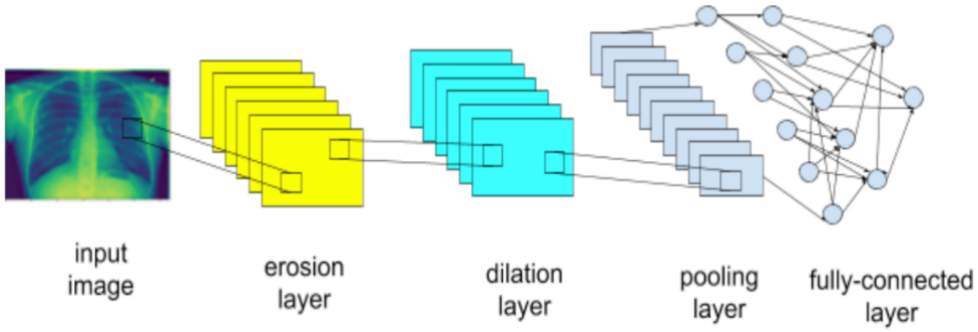
The basic morphological neural networks using morphological layers are shown in Figure 3.3 (a) shows the structure of MNN model performing erosion operation. Figure 3.3 (b) shows the structure of MNN model performing dilation operation. Figure 3.4(c) and 4(d) show the structure of MNN models performing opening and closing operations, respectively. Figs. 4(e) and 4(f) show the structure of MNN models performing top-hat and bottom-hat operations.



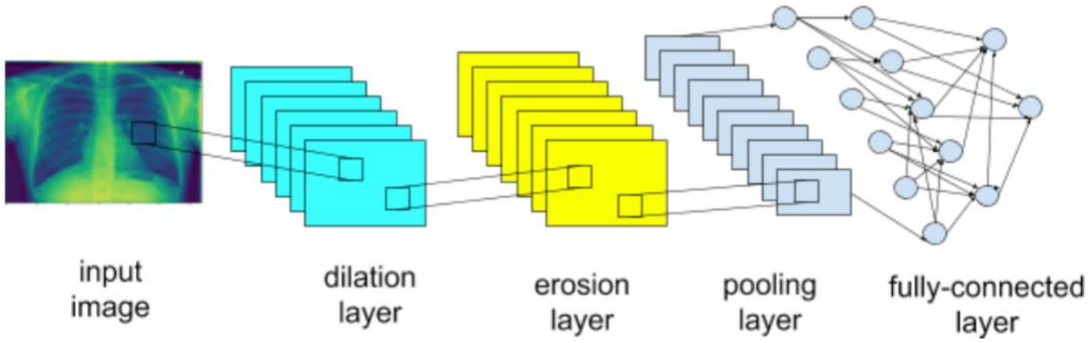
(a) Erosion classifier for pneumonia chest X-ray images



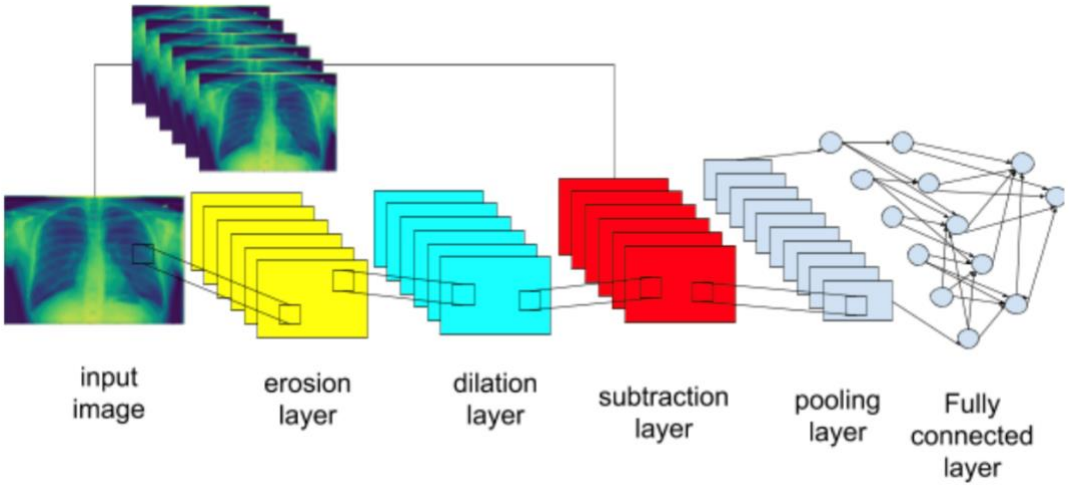
(b) Dilation classifier for pneumonia chest X-ray images



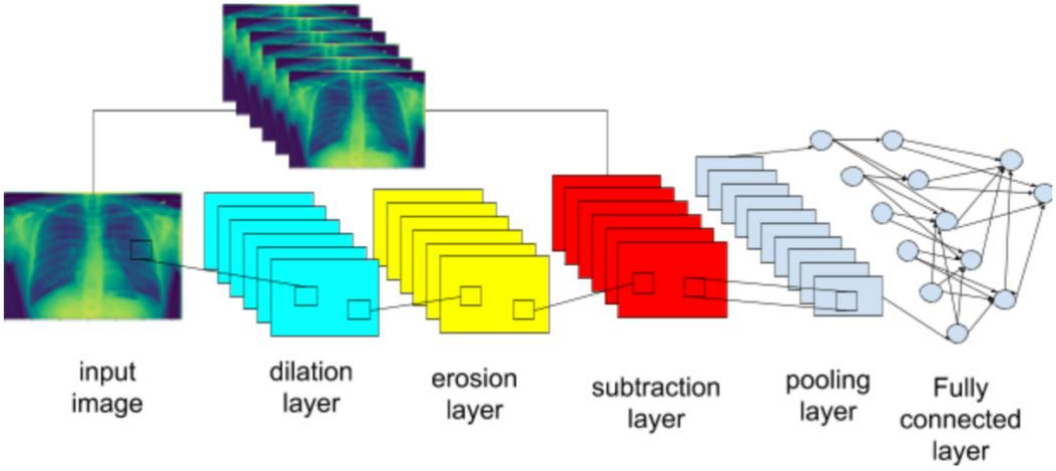
(c) Opening classifier for pneumonia chest X-ray images



(d) Closing classifier for pneumonia chest X-ray images



(e) Top-hat classifier for pneumonia chest X-ray images



(f) Bottom-hat classifier for pneumonia chest X-ray images

Figure 3.3. Morphological neural network structures for basic mathematic morphological operations.

3.2.2 Adaptive Morphological Neural Networks

Morphological operations can be various due to different combinations of dilations and erosions. From Eqs. (6) and (7), the only difference between dilation and erosion layers is the sign before the weights. Therefore, a trainable weight for sign function is used to decide the morphological operation types (dilation or erosion). The proposed adaptive morphological layer is defined in equation (3.4).

$$z_j = sign(a) * \ln\left(\sum_{i=1}^{n} e^{sign(a)*\omega_i x_i}\right) + b \tag{3.4}$$

a is an extra trainable variable aside with ω_i and b. If $sign(a)$ is $+1$, the adaptive morphological layer carries out a dilation operation layer; however, if $sign(a)$ is -1, the adaptive morphological layer carries out an erosion operation.

However, the sign function cannot be used in a deep neural network since it is not continuous making Eq. (8) undifferentiable.

To solve the undifferentiability problem, an improved sign function in the interval $[-1, +1]$ is applied for the adaptive morphological layer. The proposed morphological adaptive layer is defined in equation (3.5).

$$Z_j = \frac{e^a - e^{-a}}{e^a + e^{-a}} \cdot \ln\left(\sum_{i=1}^{n} e^{\frac{e^a - e^{-a}}{e^a + e^{-a}}\omega_i x_i}\right) + b. \tag{3.5}$$

With the proposed sign function, the adaptive morphological layers can self-learn a morphological type: dilation or erosion. A novel structure is proposed to decide the most suitable depth of the adaptive layer for pneumonia classification. Fig. 5 shows the structure of the proposed stacked adaptive morphological deep learning model. The activation functions are added before each pooling layer. After the pooling layer, the feature maps are processed by a fully connected layer and output the class predictions. The design is intended to decide the best depth for stacked adaptive layers.

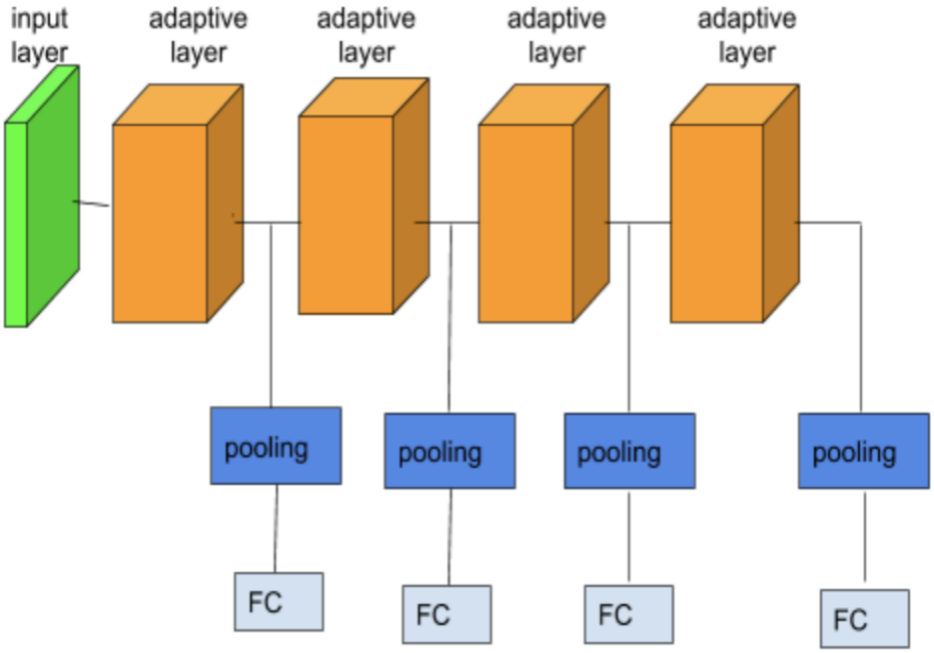


Figure. 3.4. Stacked Adaptive Morphological Deep Leaning Model.

3.3. Medical Datasets

To evaluate the performance of the proposed models, two datasets of the chest X-ray images are used. We compare the experimental results against three existing models, including LeNet, VGG16, and ResNet-50.

Two datasets are used to evaluate the performance: the chest X-Ray dataset [30] and the COVID-19 dataset [31]. The chest X-ray dataset is from Kaggle competition, which contains two categories (pneumonia/normal). It consists of 5,863 X-ray images, where 4,398 images are used for training, 1,375 images are used for testing, and 93 images are used for validation. In order to balance the training sample, we apply data augmentation in the training process.

The COVID-19 dataset contains 219 positive cases and 1,341 normal cases, where 165 positive cases and 1,005 normal cases are randomly selected in the training process. For the test dataset, 43 positive samples and 43 normal samples are used. The validation dataset contains 11 positive samples and 68 normal samples. To balance the cases in the training process, each category is augmented to 10,000 new images using image augmentation techniques. In the experiment, all the images are resized to 256×256,

3.4 Experimental Results

Table 3.1 and Table 3.2 show the experimental results of the basic morphological neural networks in two datasets. The erosion classifier and the dilation classifier use only one layer for feature extraction. In comparison, the erosion classifier achieves a 95.27% accuracy rate for the chest X-ray dataset, while the dilation classifier achieves a test accuracy rate at 98.10%. The reason is that the erosion classifier tends to shrink the images. The performance for opening and closing are similar since both operations tend to eliminate the noise. The definition for recall, precision and accuracy are defined in equation (3.6) equation (3.7) and equation (3.8).

$$recall = \frac{True\ Positive}{True\ Positive + False\ Negative} = \frac{True\ Positive}{Total\ Active\ Positive} \tag{3.6}$$

$$precision = \frac{True\ Positive}{TruePositive + False\ Positive} = \frac{True\ Positive}{Total\ Predicted\ Positive} \tag{3.7}$$

$$Accuracy = \frac{True\ Positive + True\ Negative}{Total\ Testing\ samples} = \frac{Correct\ Prediction}{Total\ Testing\ samples} \tag{3.8}$$

Table 3.1. Test Accuracy for Basic MNN in Chest X-Ray dataset

Chest X-Ray dataset	Recall	Precision	Accuracy	Total Parameter
Erosion	95.7%	96.06%	95.27%	0.81 Million
Dilation	98.21%	98.47%	98.10%	0.81 Million
Closing	98.85%	98.35%	98.41%	0.82 Million
Opening	98.60%	98.09%	98.10%	0.82 million
Top-hat	98.22%	98.01%	97.89%	0.83 Million
Bottom-hat	97.21%	96.60%	96.45%	0.83 Million

Table 3.2. Test Accuracy for Basic MNN in COVID-19 Dataset

COVID-19 dataset	Recall	Precision	Accuracy	Total Parameter
Erosion	95.23%	93.02%	94.71%	0.81 Million
Dilation	95.35%	95.35%	96.26%	0.81 Million
Closing	95.45%	97.67%	96.57%	0.82 Million
Opening	93.33%	97.67%	95.97%	0.82 million
Top-hat	93.18%	95.34%	95.15%	0.83 Million
Bottom-hat	95.23%	93.02%	94.79%	0.83 Million

Table 3.3 shows the test accuracy of the stacked adaptive morphological neural network model. We observe that the best performance for the stacked adaptive morphological neural network is achieved at six layers. An obvious overfitting occurred when the seventh adaptive layer is stacked. For the chest X-ray dataset, the best performance is 98.75%, and for the COVID-19 dataset, the best performance is 97.33%.

Table 3.3. Test Accuracy Stacked Adaptive MNN Model

Stacked Numbers	Chest X-Ray dataset	COVID-19 dataset	Total Parameter
1	75.13%	75.43%	0.81 Million
2	80.35%	84.66%	0.81 Million
3	89.41%	91.19%	0.82 Million
4	93.02%	94.97%	0.82 million
5	97.39%	95.97%	0.83 Million
6	98.75%	97.33%	0.84 Million
7	96.10%	95.10%	0.85 million
8	93.16%	92.15%	0.88 million
9	90.33%	90.26%	0.9 million

Table 3.3 shows the comparison of our proposed models against three CNN models, including LeNet, VGG16, ResNet-50, DenseNet, SqueezeNet, MobileNet and Inception v4. We observe that the proposed MNN models achieve similar and even better performance than the CNN models. Although as comparing to the best performance Inception v4 model, the proposed model achieves the highest performance at 98.75% and 97.33%, the total of parameters in the proposed model is

reduced by 98.7% significantly against the parameters in Inception v4 model. Even compared with the CNN model has the least parameters (SqueezeNet), our proposed model could achieve better performance.

Table 3.4. Comparison with CNN Models

Model	Chest X-Ray dataset	COVID-19 dataset	Total Parameter
The proposed stacked adaptive MNN	98.75%	97.33%	0.84 Million
LeNet [1]	85.92%	79.68%	1.4 Million
VGG16[8]	95.77%	93.27%	9.1 Million
ResNet[9]	98.69%	96.78%	25.6 million
DenseNet[14]	98.91%	97.44%	30.2 million
SqueezeNet [32]	90.53%	90.26%	0.49 Million
MobileNet [33]	91.02%	92.21%	4.2 Million
Inception v4[12]	99.04%	97.77%	65 Million

3.5 Conclusion

In this chapter, the morphological neural networks are used for the classification tasks for chest X-ray images. Traditional deep learning models such as CNN contains a giant number of parameters in the feature extraction process to achieves a high performance. The MNN models could achieve a similar result with far more less parameters than the CNN models. This advantage makes MNN more competitive than CNN models to deploy in website or other platforms. Two deep learning models are introduced in this chapter. In the basic morphological neural network, the operation type needs to be specified before training. The adaptive morphological neural network is able to train a sign function to help the model to self-learn the morphology operation type. Experimental results show MNN models can achieve better performance with much less parameters in chest x-ray datasets. Considering the effectiveness for MNN models in classification task, the MNN models is able to be applying such model to other computer vision tasks, such as image segmentation or objective detection.

Chapter 4

JOINT TASK LEARNING MODEL FOR PNEUMONIA CLASSIFICATIONN AND SEGMENTATION ON MEDICAL IMAGES

Chest X-ray images are notoriously difficult to analyze due to the noisy nature. Automatic identification of pneumonia on medical images has attracted intensive study recently. In this paper, a novel joint-task architecture that can learn pneumonia classification and segmentation simultaneously is presented. Two modules, including an image preprocessing module and an attention module, are developed to improve both classification and segmentation accuracies. Experimental results performed on the massive dataset of the Radiology Society of North America have confirmed its superiority over other existing methods. The classification test accuracy is improved from 0.89 to 0.95, and the segmentation model achieves an improved mean precision result from 0.58 to 0.78. Finally, two weakly supervised learning methods: class-saliency map and grad-cam, are used to highlight corresponding pixels or areas which have significant influence on the classification model, such that the refined segmentation can focus on the correct areas with high confidence.

4.1 Baseline Model

In this section, the original joint-task learning model for classification and segmentation is presented. The model performs binary classification that separates pneumonia samples from healthy ones. The classifier is based on VGG16 and contains three parts: the input layer, feature extraction layers, and fully connected layers. The loss function using binary cross-entropy is defined in equation (4.1).

$$BCE\_Loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \ log\ log\ \left(p(y_i)\right) + (1 - y_i)\ log\ log\ \left(1 - p(y_i)\right) \quad (4.1)$$

y_i is the label (1 for pneumonia pixel and 0 for healthy pixel) and $p(y_i)$ is the predicted probability of the pixel belonging to pneumonia for all N pixels. In the segmentation task, the model is required to output a pixelwise label map, where the target area is labeled as 1 while other areas as 0. The segmentation model is an encoder-decoder structure. The encoder converts an input image x into a latent-space representation h as $h = f(x)$. The decoder reconstructs the input from latent space representation h to a label map r is defined in equation (4.2)

$$r = g(h). \quad (4.2)$$

The autoencoder is defined in equation (4.3).

$$r = g(f(x)). \quad (4.3)$$

By encoding the input image into latent representation and decoding it back to a label map, each pixel is assigned a label in the reconstruction process. Pixels labeled as 1 represent belonging to an opacity area, while the normal area is labeled as 0.

The segmentation model is a U-net like structure. The loss function in our segmentation model uses mean square error, which can be described as the summation of squared distances between ground truth map and decoded label map. Let y_i represent the ground truth for i-th pixel and Y_i represent the model's prediction for i-th pixel. The mean square error loss is computed in equation (4.4)

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - Y_i)^2 \quad (4.4)$$

The baseline joint-task learning model combines the classification and segmentation models with sharing feature extraction layers. The original joint-task learning model is shown in Figure 4.1. An input image is firstly going through convolutional layers for feature extraction. Secondly, the feature maps are fed into dense layers for classicization and output the class types: Pneumonia or Healthy. At the same time, the feature maps are fed into the decoder for segmentation. Finally, in the segmentation model, the feature maps in the first step are concatenated with the feature maps and output the segmentation maps.

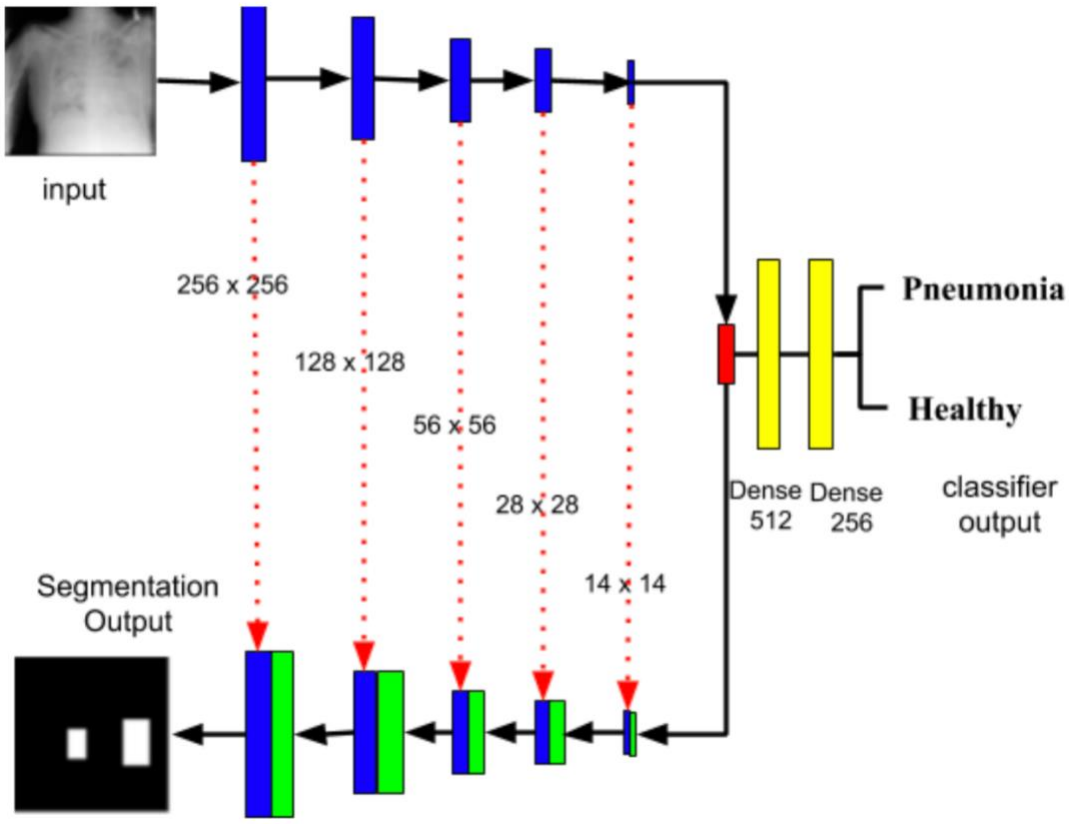


Figure 4.1. The Original Joint-Task Learning Model.

4.2 Class Saliency Map and Grad-CAM

When the training of the joint-task learning model is finished, a class saliency map [41] and a Grad-Cam [42] are used to interpret the classifier and visualize the corresponding area which has a great influence. A high-class score means a relatively high influence. The class saliency maps compute the class score $S_c(I)$ from a given test image I in equation (4.5)

$$S_c(I) = \ w_c^T I + b_c \tag{4.5}$$

where the label for image I is c. The class score's derivative w is defined in equation (4.6)

$$w = \frac{\partial S_c}{\partial I} \tag{4.6}$$

By computing w in back-propagation, the pixels which have a stronger influence in determining class-score can be found. Thus, the class saliency map is determined by the classification model and class c. By visualizing the corresponding saliency map, one can understand why the classification model makes such a decision. Although the class saliency map is not a restrict segmentation tool, especially in lung CT images, it can still highlight corresponding pixels.

The grad-cam or gradient-weighted class activation mapping performs a weakly supervised localization according to the image's label and the gradient of the model's last convolutional layer. For a given image and its label, the image is forward-propagated to the CNN model, and a confidence score is obtained for its corresponding label. The signal is then back-propagated to produce the feature maps. Finally, a ReLU

activation function is used to combine the feature maps to show where the model is focused on when the prediction is made. Compared to CAM [43], the Grad-cam is a generalization method and can be applied to any CNN model without modifying the model's structure. By visualizing the testing samples of using class saliency map and grad-cam in different models, it is possible to visualize whether the model focuses on the correct area or not.

4.3 Image Preprocessing and Visual Attention Modules

In this section, the image preprocessing and visual attention module is discussed. The purpose for this module is to improve the baseline model's performance and remove noise in the original dataset.

4.3.1 Image Preprocessing Module with Morphological Layers

Mathematical morphology is a widely-used approach for shape representation and image preprocessing in image processing. Two fundamental morphological operations are dilation and erosion. Let the input image be I and the structuring element be s. Dilation is denoted as $I \oplus s$, which expands the image by the structuring element. Erosion is denoted as $I \ominus s$, which shrinks the image by the structuring element.

The opening is typically used for contour smoothing, especially for breaking thin connections between components and enlarging small holes or gaps. It is defined as an erosion followed by a dilation as

$$I \circ s = (I \ominus s) \oplus s \tag{4.7}$$

Different from opening, the closing can be used for connecting narrow areas and filling in small holes or gaps. It is defined as a dilation followed by an erosion as

$$I \bullet s = (I \oplus s) \ominus s \tag{4.8}$$

Figure 4.2 shows two sample images from the Kaggle Pneumonia dataset, which are processed using dilation and erosion with a 6×6 structure element of all 1's. Fig. 3 shows two sample images, which are processed using closing and opening with a 6×6 structure element of all 1's.

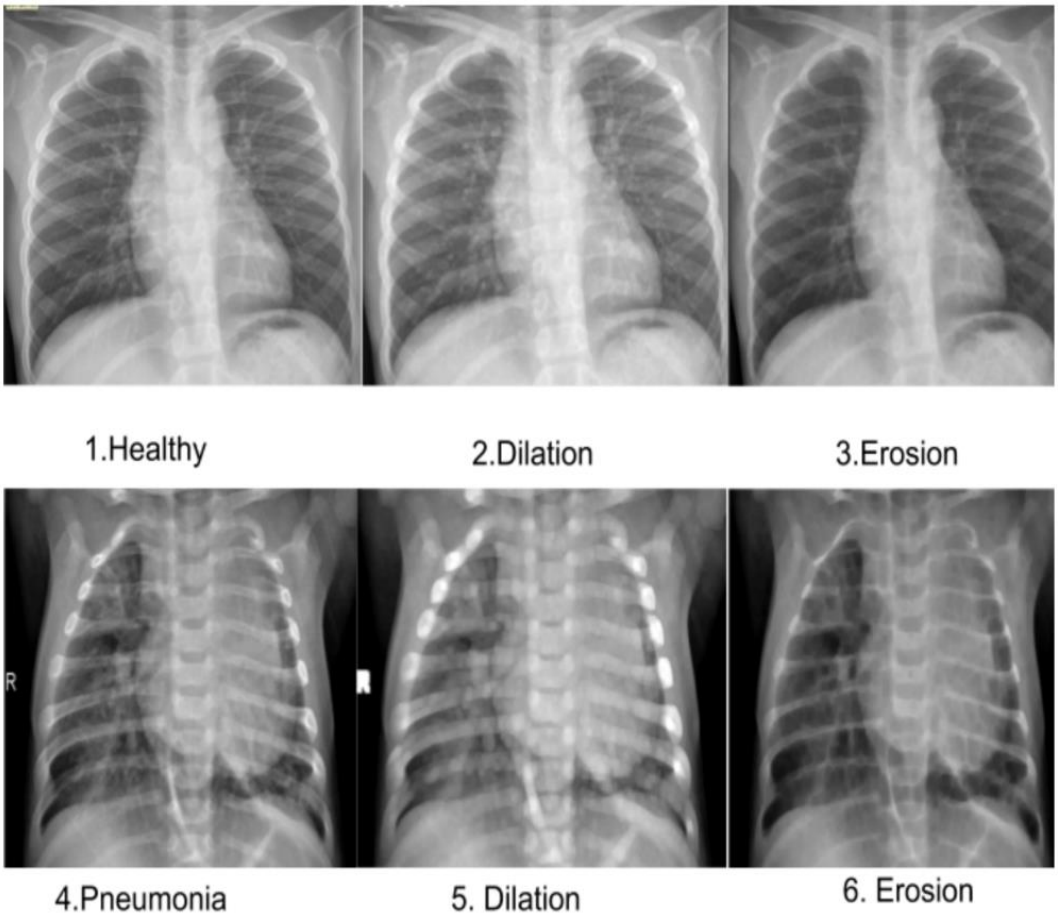


Figure 4. 2. Sample images after morphological operations. Column 1 shows input images; column 2 shows dilation; column 3 shows erosion.

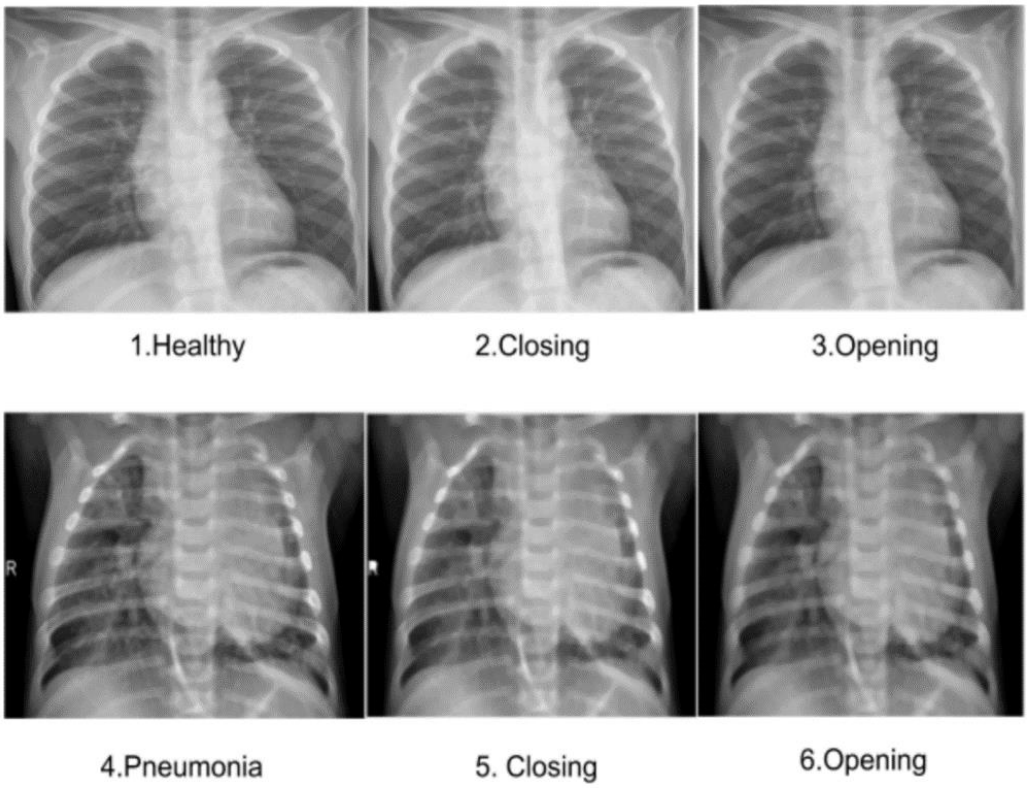


Figure 4.3 Sample images after morphological operations. Column 1 shows input images; column 2 shows closing; column 3 shows opening

Previous work on morphological neural network [45] is applied as preprocessing and a feature extraction layer is used for classification. Dilation can expand some of the small areas while enlarging some of the noisy areas. Erosion can clean the background by eliminating some noisy areas, but at the same time, filtering out some pixels. Opening and closing can smooth the contour, where closing tends to fill in some holes and opening tends to make them larger. Figure 4.4 shows four basic morphological operations using morphological layers.

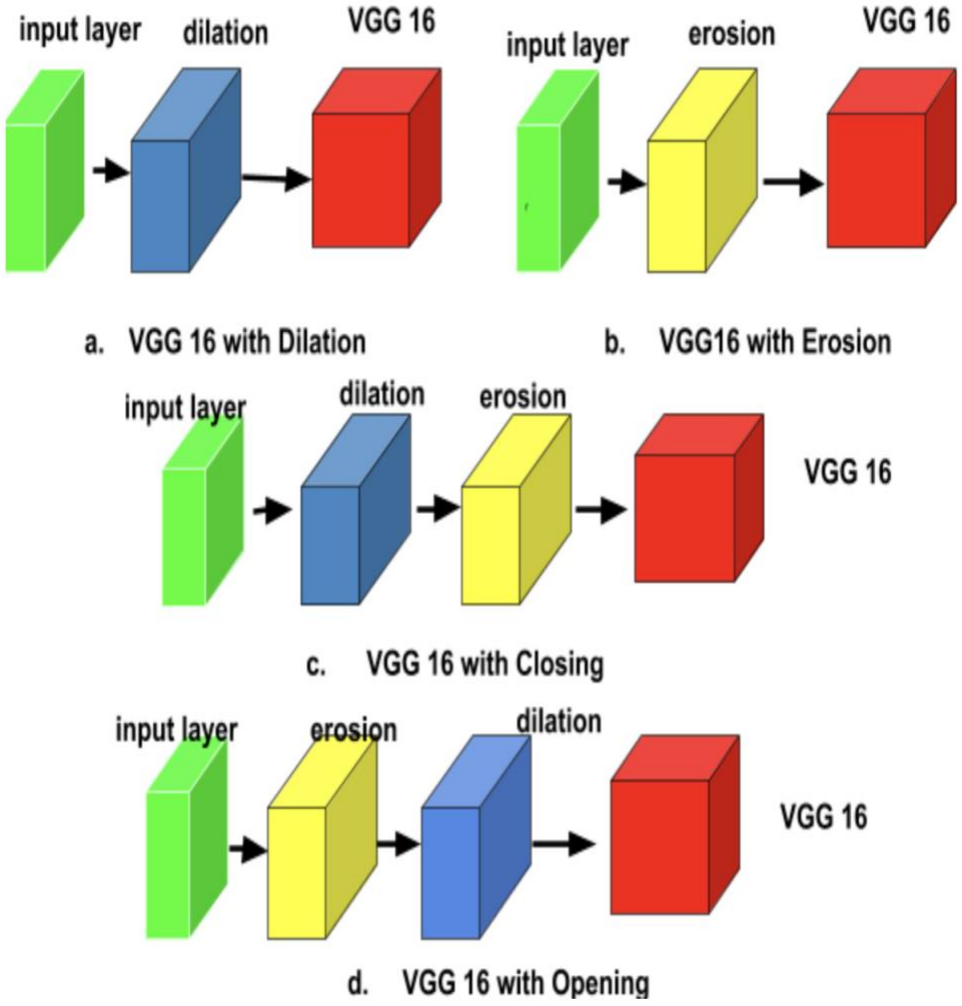


Figure 4.4. Morphological image preprocessing modules with morphological operations.

4.3.2 Visual Attention Modules

The convolutional block attention module (CBAM) [44] and morphological block attention module (MBAM), are applied separately to improve the performance of the original joint-task learning model. The CBAM is used to learn the weight of feature maps in convolutional layers. While the MBAM is used to learn the weight of feature maps in morphological layers and to refine the feature maps between morphological layers and correctly locate a target area. The two visual attention modules are shown in Figure 4.5.

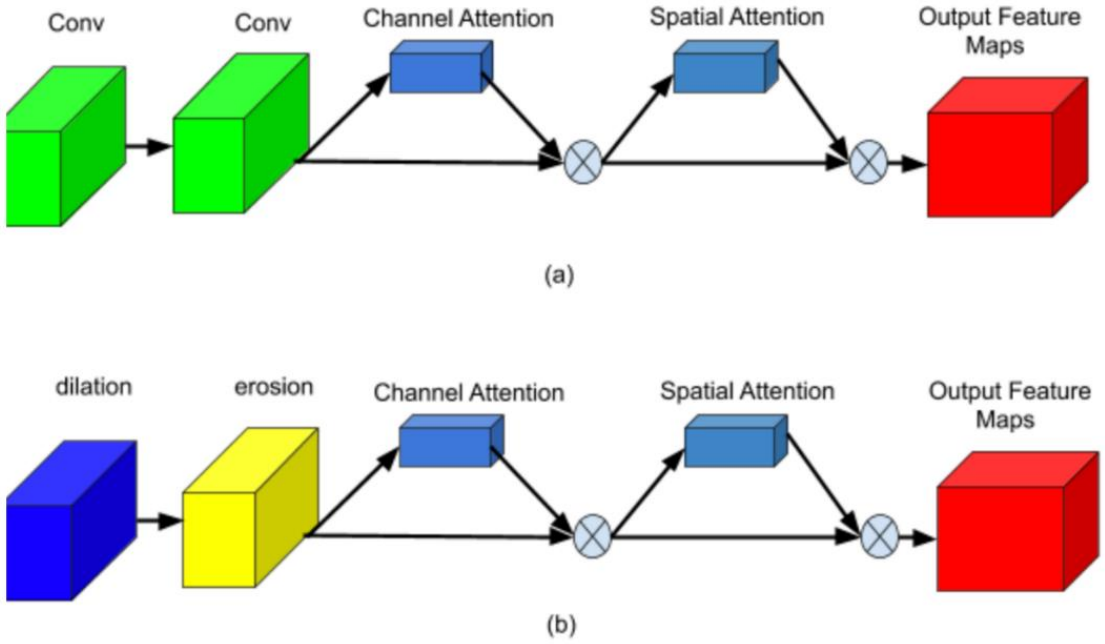


Figure 4.5. Visual attention modules (a) Convolutional block attention module, (b) morphological block attention module.

4.4 Experimental Results

Experiments of combining different modules with the proposed joint-task learning model are conducted in this section. In the segmentation task, a U-Net like structure is used for reconstructing the masks. Considering that the ground truth is given by a bounding box instead of pixelwise label maps, performing a pixelwise segmentation may encode non-opacity regions inside a bounding box and further influence the model's prediction. The bounding box may indicate a rough area containing the lung opacity but cannot annotate each pixel. The segmentation model may not be able to preciously recognize a target area. Thus, we evaluate the performance of the joint-task learning model by showing both the segmentation model and the weakly supervised segmentation result.

The dataset from Kaggle's RSNA (Radiological Society of North America) Pneumonia Detection Challenge [46] is used, which contains CT chest images in the DICOM format. The pixel in the opacity area is labeled as 1, indicating a potential pneumonia sample; otherwise, it is labeled as 0. Figure 4.6, (a) shows an image which does not contain the opacity area Figure 4.6 (b) shows an image containing two opacity areas. The dataset contains 9,555 samples with pneumonia and 8,851 normal (healthy) samples. This dataset is randomly shuffled and divided into three groups: training data, validation data, and testing data, which respectively have 13,804 (75%), 920 (5%), and 3,862 (20%) images. To compare the performance of each model, all the experiments conducted in this research uses the same images for training, validation and testing.

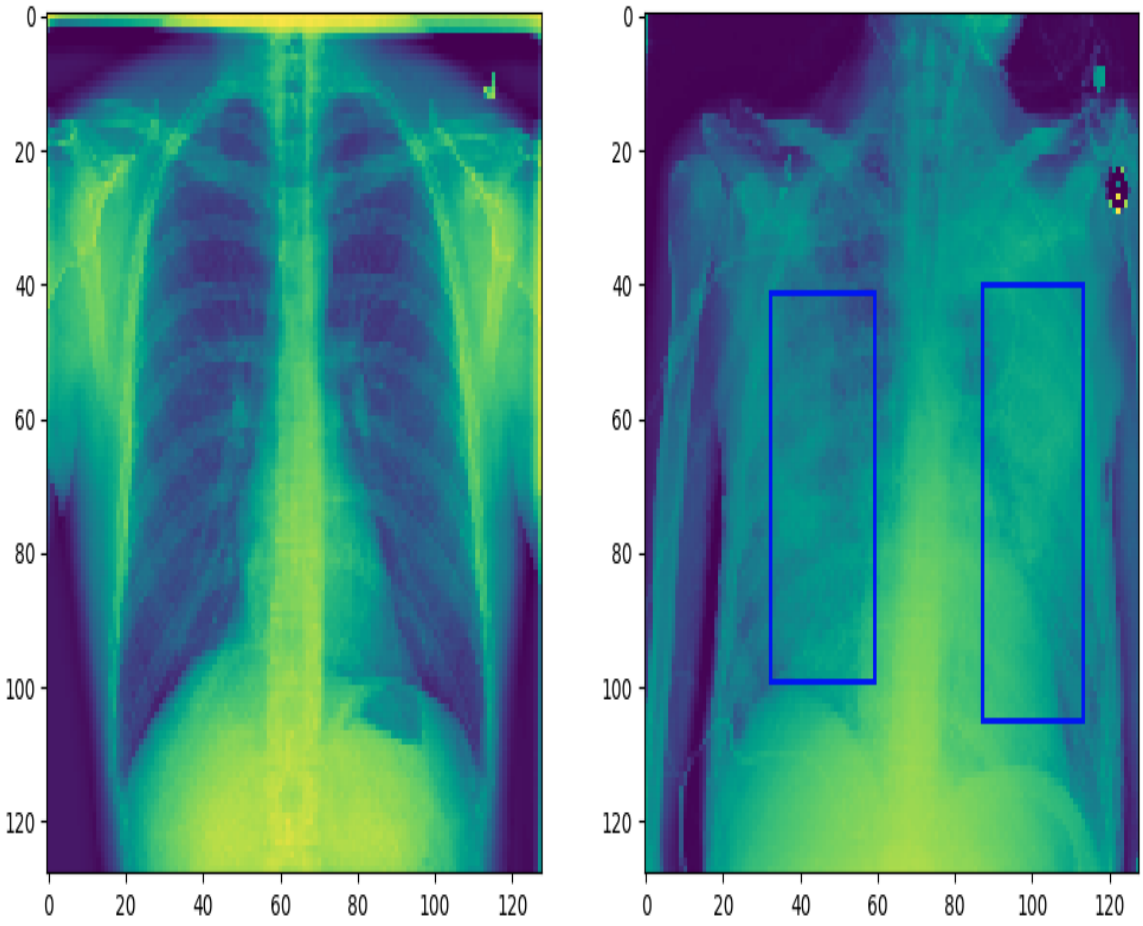


Figure 4.6. Sample images in RSNA Pneumonia Detection Challenge. (a) Healthy body (b) sample with lung opacity.

4.4.1 Performance of the Baseline Joint-task Learning Model

To design the proposed joint-task learning model, two main problems need to be solved. First, it is difficult for the classification and segmentation models to converge at the same time. The reason is the classification model converges much faster than the segmentation model. In the segmentation model, the decoder part has similar parameters with the encoder part, which is far more overweight than the parameters in classification model. Second, the parameter in the convolutional layers should be sufficient to extract the features and cannot be overweighed due to the limited computational capacity. Thus, the classification model uses a VGG16 structure and the segmentation model use a U-Net structure.

The joint-task learning model is compared against different models. For classification, it is compared with ResNet-50, and for segmentation, it is compared with SegNet, FCN and DeepLab V3 [47]. The performance of these models is listed in Table 5.1

Table 4.1. Test Accuracy for Original Joint-Task Learning Model

Model	Classification Accuracy	Classifier Parameter	Segmentation MAP	Total Parameter
Joint-task Model	89.27%	9.1 Million	0.5945	25 Million
SegNet	/	/	0.5072	21.8 Million
FCN	/	/	0.4368	9.1 Million
ResNet-50	88.73%	25.6 million	/	25.6 million
Deep Lab V3 [47]	/	/	0.6012	2.5 Million

For classification, VGG16 and ResNet achieve a similar test accuracy. Our proposed joint- task learning model, FCN, and SegNet use a VGG16 as feature extractor. However, in the up-sampling part our joint-task learning model uses a U-Net structure, which adds the corresponding feature maps from previous feature extractors. Compared to FCN and SegNet, our proposed joint-task learning model can directly combine previous feature maps in the feature extraction process to achieve a higher mean-average precision. When compared with the most recent semantic segmentation model-- the Deep Lab V3 [36], our joint-task learning model can achieve similar performance. Since the ground truth is just a roughly area with a bounding box, it is hard for the segmentation models to recognize each pixel precisely. Although Deep lab V3 has less parameters and a better performance, it cannot perform the classification task.

4.4.2 Performance of the Different Joint-Task Learning Models

The baseline model classifier utilizes a VGG16 structure, which is combined with different modules: morphological layers, CBAM, and MBAM. Table 4.2 shows different combinations of morphological layers as a pre-processing module with a VGG16 classifier on the Kaggle pneumonia dataset. The performance of CNN classifier works as a baseline model and achieves a accuracy at 89.13%. It is observed that the opening + closing + VGG16 model achieves a relatively high-test accuracy. In Figure 4.2, it is clear to find a dilation can blur the CT image, while an erosion can clear the noise. The pre-processing module using a dilation layer has a relatively weak performance than the erosion layer + CNN model. The opening and closing operations are both designed for contour smoothing. The better performance for the image preprocessing module is through two different smoothing layers, which add more smoothing, so the infected samples are easier to be recognized.

Table 4.2 Test Accuracy for Classification Accuracy Different Morphological Layers

Model	Classification Accuracy
VGG16	89.13%
Dilation + VGG16	88.38%
Erosion + VGG16	91.62%
Closing +VGG16	93.02%
Opening+VGG16	92.78%
Opening + Closing + VGG16	94.32%
Closing + Opening+ VGG16	94.14%

Figure 4.7 shows the proposed models, where (a) VGG16 model, (b) the structure of morph layers + VGG16, (c) the structure of CBAM + VGG16, (d) the structure of Morph layers + CBAM + VGG16, and (e) the structure of MBAM + CBAM + VGG16.

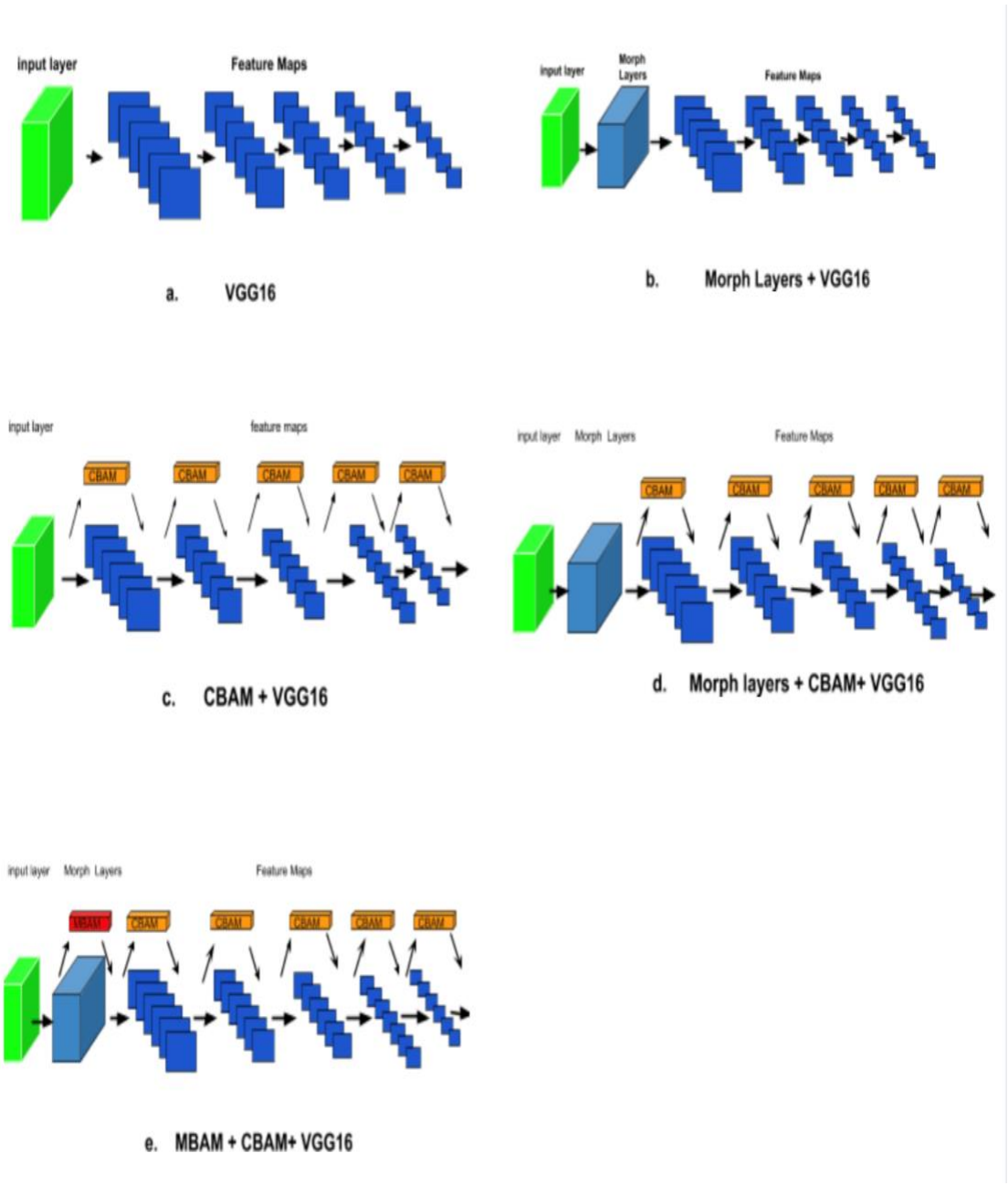


Figure 4.7 The Proposed Joint-task Learning Models.

The performance of the proposed joint-task learning model is listed in Table 3. As compared to the baseline model, the MNN + VGG16 model achieves a 5.13% improvement in classification and 2.32% improvement in segmentation. The reason for this improvement is caused by the image pre-processing layers using morphological layers. The MNN layers use soft minima or soft maxima function to respectively approximate dilation or erosion, which mathematically performs the morphological filtering on input images to enrich the feature maps.

Table 4.3 Test Accuracy for Joint-task Learning Model with Different Modules

Model	Classification Accuracy	Segmentation MAP
VGG16	89.27%	58.45%
MNN+ VGG16	94.14%	60.73%
CBAM + VGG16	93.85%	71.78%
MNN+CBAM+VGG16	90.85%	63.85%
MBAM+CBAM+VGG16	95.73%	78.72%

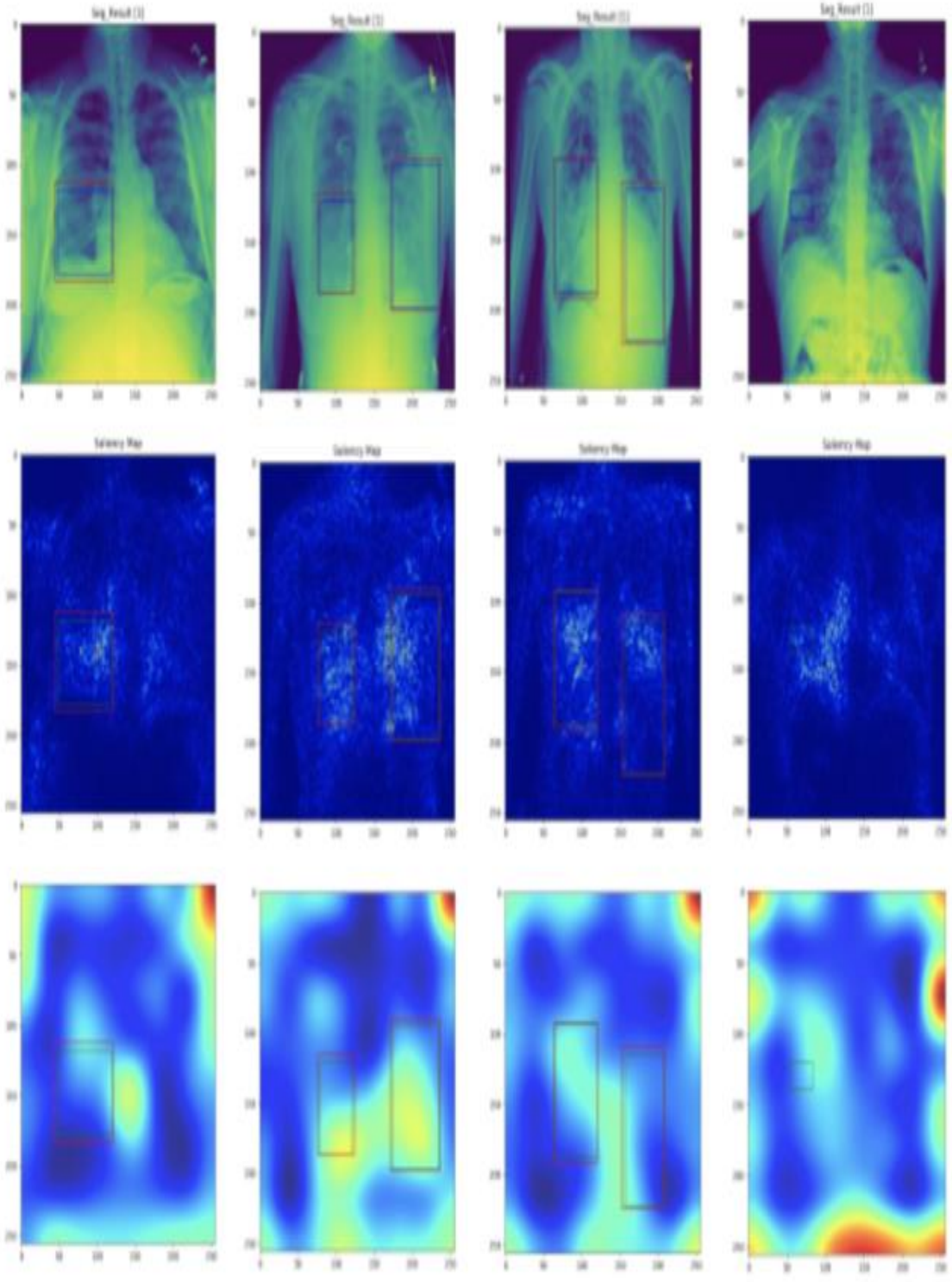
The CBAM+VGG16 model utilizes the CBAM mechanism to refine the feature maps between convolutional layers and improves the classification model by 4.58% and the segmentation model by 13.33%. The reason for this improvement is that CBAM guides the model in both spatial domain and channel-wise domain.

The MNN + CBAM + VGG16 model combines MNN and CBAM. Even though the classification rate is increased by 1.58% and the segmentation MAP is increased by 5.4%, it is still worse than MNN + VGG16 and CBAM + VGG16. The reason is that MNN layers and CBAM change the gradients in original images.

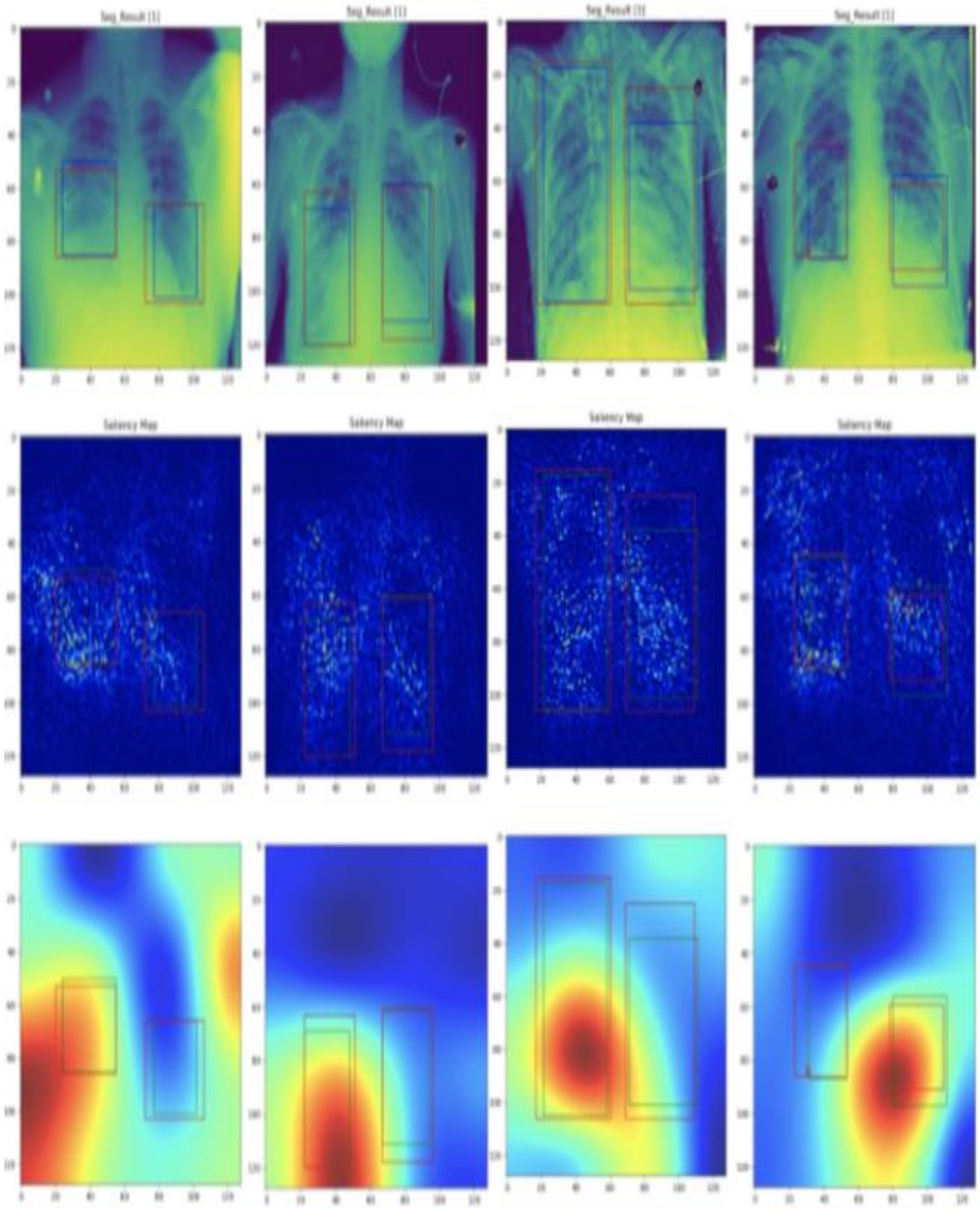
The MBAM + CBAM + VGG16 model refines the feature maps between convolutional layers and between morphological layers. Experimental results show that it improves the classification accuracy by 6.46% and the segmentation by 20.27%, as compared to the baseline model. The MBAM correctly guides the MNN layers in the training process to correct the gradient in MNN + CBAM + VGG16, where the gradient is changed due to unorganized feature maps in morphological layers.

4.4.3 Evaluate Model Performance by Class Saliency Map and Grad-Cam

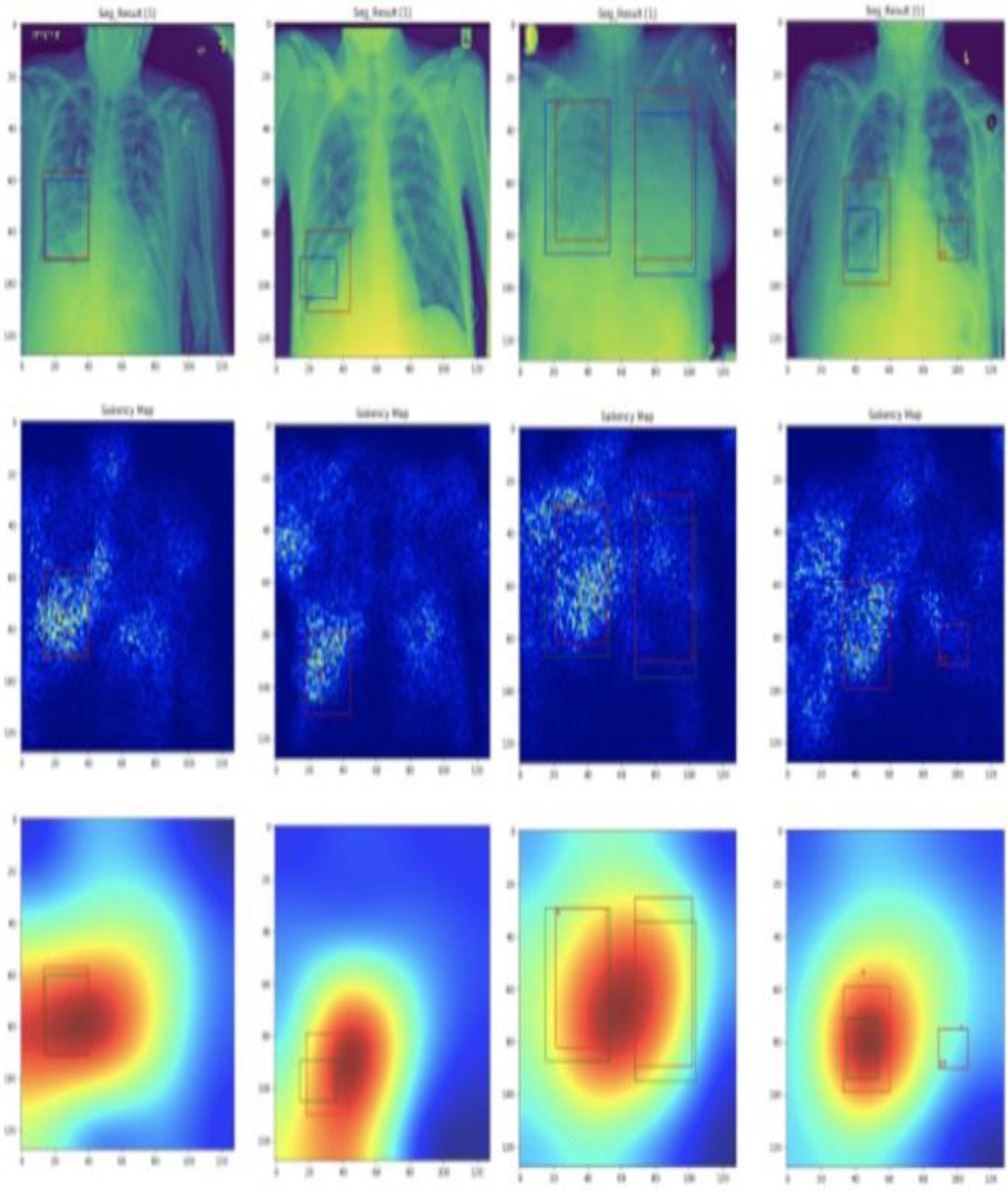
The class saliency maps and Grad-Cam on four random samples from the test dataset to illustrate the model performance. Since the original joint-task learning models have confidence ranging from 89% to 95%, it is critical to interpret whether the classifiers can detect the correct area. The class saliency map shows the corresponding influential pixels when the classifier makes its prediction. The Grad-Cam shows the probability map to indicate which area has a high possibility when the classifier makes the prediction. By attaching the segmentation model's prediction with bounding boxes, we can finally decide whether this model is trusted. Fig. 8 shows different model's performance on four pneumonia samples. The first row shows the segmentation prediction in a red bounding box, while the ground truth is displayed as a blue bounding box. The second row shows the class saliency map, and the third row shows the Grad-Cam attention map.



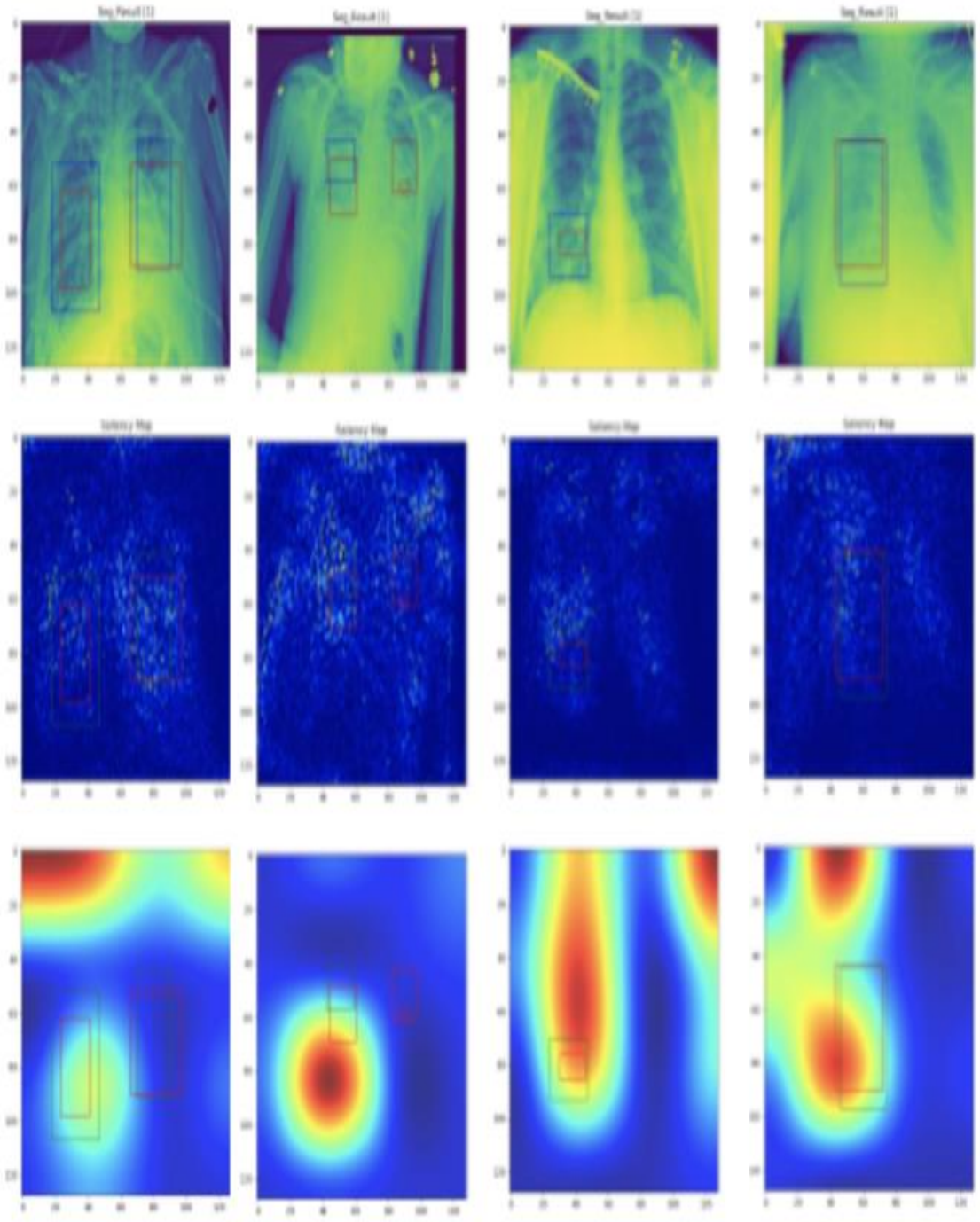
a. Baseline Model



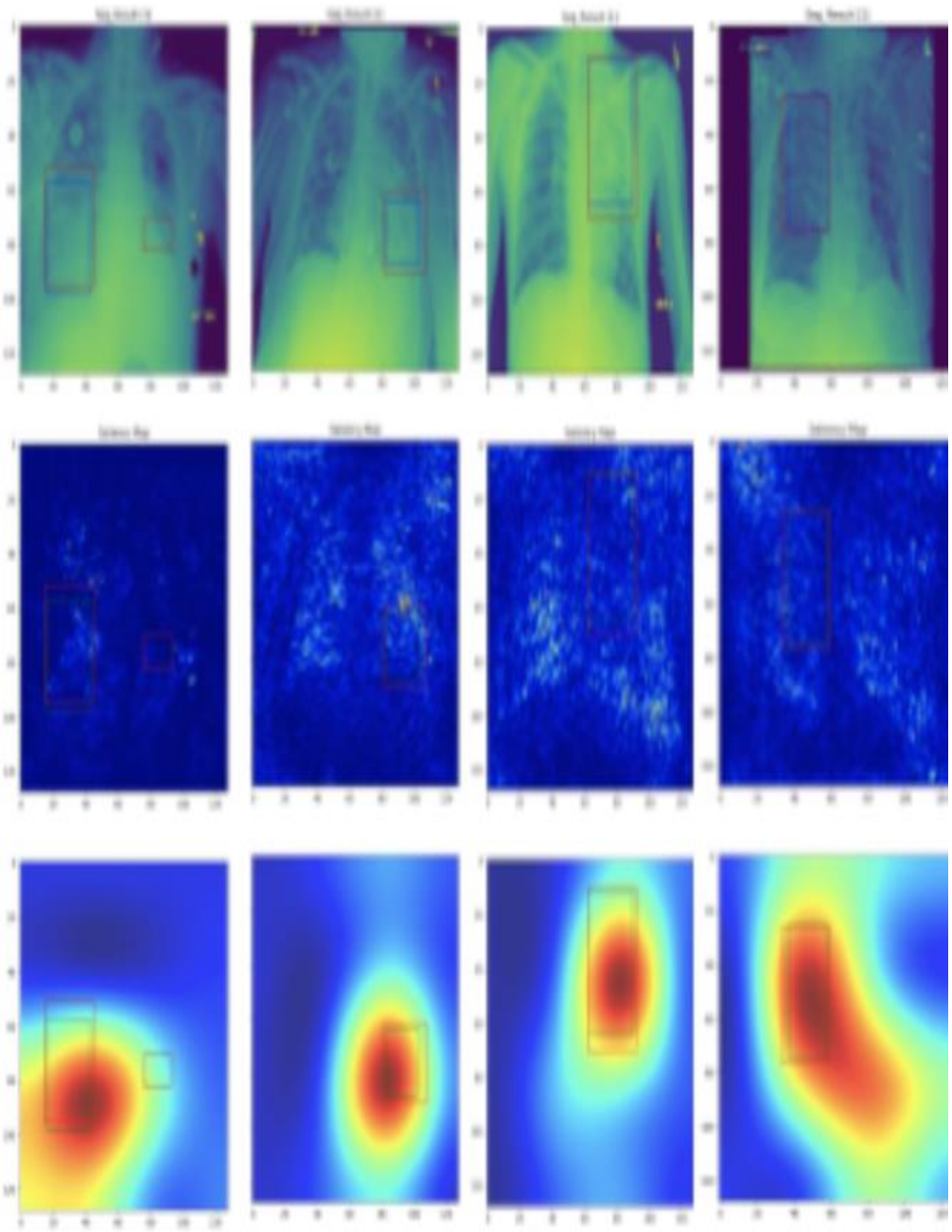
b. Baseline Model + MNN(closing + opening)



c. CBAM + Baseline Model



d. MNN + CBAM + Baseline Model



e. MBAM+ CBAM + Baseline Model

Figure 4.8. Class saliency map and Grad-cam for different models.

Figure 8(a) shows that the samples are all classified as pneumonia. The class saliency map shows a weak segmentation of the lung area. The Grad-Cam maps show that the baseline model is more likely to focus on the corners or bottom, instead of the lung area when making its prediction. The target area has a relatively low attention probability. Thus, the baseline model has poor performance because the classifier makes its prediction based on the wrong attention area.

Figure 8(b) shows the baseline model with morphological layers. The class saliency map shows possible influential pixels. The morphological layers improve the model to focus on the correct attention area, so the Grad-Cam can focus on the target area instead of other areas of the test images in the baseline model. Fig. 8(c) shows the samples for the baseline model with convolutional block attention module, which successfully improves the baseline model by channel-wise attention and spatial attention modules. Compared to the baseline model, the CBAM guides the model to focus on target areas correctly.

Figure 8(d) shows the samples for the baseline model combined with morphological layers and CBAM. Since the morphological layers are not well guided, the image preprocessing module misleads the model to focus on other areas. Fig. 8(e) shows the samples for the baseline model combined with MBAM and CBAM. Compared to the Grad-Cam maps in Fig. 8(d), the morphological layers are well guided by attention modules. Thus, the model can focus on the correct target with higher confidence and solve the problems as shown in Figure 8(a) and Figure 8(d).

4.5. Conclusion

In this chapter, a joint-task learning model is proposed for pneumonia classification and segmentation. The effectiveness of this model is proven by comparing different classification or segmentation models. From visualizing the class saliency map and Grad-Cam map, we find that the baseline model's classifier focuses on other areas instead of the target area. The image preprocessing and attention modules are developed to refine the joint-task learning model. Experimental results show that the CBAM or the morphological layers can help the proposed joint-task learning model to focus on the correct area with higher confidence. Furthermore, by combining the MBAM and CBAM to the baseline model, the proposed joint-task learning model not only achieves the best classification test rate at 95.73% and the best mean-average precision of 0.7872, but also helps the classification model to focus on the correct area.

Chapter 5

THE ATTENTIONED MORPHOLOGICAL AND CONVOLUTIONAL NEURAL NETWORK FOR ECOLOGY DATA AND MEDICAL IMAGE

5.1 Morphological Neural Networks in Ecology Datasets

In section 3 and section 4, the morphological neural networks are used for different tasks. In the previous chapters of this research, the ecology datasets (bee wings and butterfly datasets) and the Chest X-ray datasets (Kaggle dataset and COVID 19 dataset) are respectively used to test on the morphological neural networks. To evaluate the performance of MNN in ecology datasets and medical datasets, experiments on all ecology datasets and medical datasets are conducted in this chapter. First, ecology datasets are used for the basic morphological operation neural networks. Table 5.1 shows the results in the bee wing dataset and butterfly dataset.

Table 5.1 shows the results in bee wing dataset and augmented bee wing dataset and Table 5.2 shows the results in butterfly dataset and augmented bee wing dataset. To compare with the performance with CNN models, the relevant experimental results are added after the MNN models. The experimental results show MNN can achieves relatively similar and even higher in some of this model. Second, the adaptative morphological neural works are used for the ecology datasets. Table 5.3 shows the test accuracy of stacked adaptive morphological neural network in Bee Wing dataset and augmented Bee Wing dataset. Table 5.4 shows the shows the test accuracy of stacked adaptive morphological neural network in the Butterfly dataset and the augmented Butterfly dataset.

Table 5.1. MNN in Bee Wing Dataset and Augmented Bee Wing Dataset

Bee Wing	Original dataset	Data Augment
Erosion	84.53%	85.64%
Dilation	86.15%	88.53%
Closing	87.76%	89.37%
Opening	87.93%	89.77%
Top-hat	87.39%	89.55%
Bottom-hat	87.41%	88.89%
LeNet-5	87.78%	89.97%
AlexNet	86.04%	89.8%
VGG16	17.74%	88.7%
VGG19	17.72%	87.34%
ResNet50	86.54%	89.34%
Inception v3	87.16%	91.46%
InceptionResNetV2	87.72%	90.91%

Table 5.2. MNN in Butterfly dataset and Augmented Butterfly Dataset

Butterfly	Original dataset	Data Augment
Erosion	67.33%	69.81%
Dilation	68.45%	70.31%
Closing	76.76%	78.53%
Opening	77.93%	79.48%
Top-hat	79.10%	81.55%
Bottom-hat	79.71%	80.89%
LeNet-5	70.24%	71.41%
AlexNet	79.85%	80.83%
VGG16	17.74%	79.91%
VGG19	17.72%	80.33%
ResNet50	79.21%	86.54%
Inception v3	80.32%	87.16%
InceptionResNetV2	81.94%	87.72%

Table 5.3. Test Accuracy Stacked Adaptive Morphological Neural Network Model

Stacked Numbers	Bee Wing dataset	Augmented Bee Wing dataset	Total Parameter
1	65.13%	68.43%	0.81 Million
2	70.55%	72.66%	0.81 Million
3	81.49%	85.19%	0.82 Million
4	**87.72**%	88.97%	0.82 million
5	87.39%	89.97%	0.83 Million
6	86.61%	**90.33%**	0.84 Million
7	84.10%	90.10%	0.85 million
8	80.16%	89.15%	0.88 million
9	79.63%	89.26%	0.9 million

Table 5.4. Test Accuracy of the Stacked Adaptive Morphological Neural Network Model

a	Butterfly dataset	Augmented Butterfly dataset	Total Parameter
1	55.33%	60.77%	0.81 Million
2	60.75%	75.66%	0.81 Million
3	73.66%	81.19%	0.82 Million
4	78.72%	83.64%	0.82 million
5	80.39%	87.30%	0.83 Million
6	**81.61%**	**88.33%**	0.84 Million
7	80.10%	85.10%	0.85 million
8	79.16%	83.89%	0.88 million
9	77.63%	82.62%	0.9 million

Compared with CNN models, the morphological neural networks contain relatively less parameters and could achieve even higher test accuracy. For the ecology datasets and chest x-ray datasets, MNN is even more affective than CNN models. However, MNN is not always surpass the CNNs. In the next section, the MNN will extend to more datasets and thoroughly evaluate the advantages and disadvantages in morphological neural networks.

5.2 The Limitations of MNN Model

MNN refers as the morphological neural network, which use mathematical morphology as a feature extraction mechanism. Compared with convolutional neural network, which uses convolution operation to amplify and extract features from image, MNN replace this process by local minimum or local maximum. MNN is proposed for different tasks, such as handwritten digits (MNIST) classification, traffic sign recognition and brain tumor sign recognition (MRI brain), geometric shapes dataset, ecology datasets and chest X-ray datasets. Also, MNNs are also used to detect other datasets such dogs and cats' datasets.

In this part, the MNN models are applied to more datasets to extend it performance on more datasets. The extended datasets including the Brain Tumor Dataset [48], the MNIST Dataset [49], the Traffic Sign dataset [50], the Geometric Shapes Dataset and the Cat and Dog dataset [51].

The Brain Tumor dataset [48], also called the MRI Brain Dataset, contains 3,064 grayscale images from 233 patients with three kinds of brain tumor: meningioma (708 samples), glioma (1426 samples), and pituitary tumor (930 samples). In the experiment, all the images are 64 × 64 for classification, and 2,910 images are used for training and 154 images for testing.

The MNIST Dataset [49] is a database consisting of 70,000 examples of handwritten digits 0~9. It has 60,000 training images and 10,000 testing images. The image size in the MNIST Dataset are all 28 × 28 grayscale images in 10 classes.

The Geometric Shapes Dataset contains 120,000 grayscale images of size 64 × 64 in 5 classes: ellipse, line, rectangle, triangle, and five-edge polygon. The images are created by randomly drawing white objects on a black background, where the size, position, and

orientation are randomly initialized. There are 20,000 images in each class for training and 5,000 images used in each class for testing.

The Traffic Sign Dataset, or named the GTSRB Dataset, introduces a single-image, multi-class classification problem, and there are 42 classes in total. The images contain one traffic sign each, and each real-world traffic sign only occurs once. We resize all the images into 31 × 35 and select 31,367 images for training and 7,842 images for testing. All the images are in grayscale. Figure 5.1 shows sample images of the following datasets.

Figure 5.1 The examples from the four datasets in the experiments. The first row is the images from brain tumor dataset, the second row from MNIST dataset, the third row from GTSRB dataset, and the fourth row from SCGS dataset.

The Cat VS Dog Dataset contains 25000 RGB images. There are 12500 image of cats and 12500 image of dogs. The training datasets contains 18750 (75% total) images and the testing dataset contains 3750 (15% total) images. To avoid overfitting in the training process, a validation dataset, which contains 1250 (5% total) images, is applied. Figure 5.2 shows the sample images in the Dog VS Cat Dataset.

Figure 5.2 The examples from the sample images Dog VS Cat Dataset in this experiment. The left part shows the sample images of cays and the right part shows the sample images of dogs.

To evaluate the performance of MNN, the comparison experiments are conducted in different CNN models. The CNN models including LeNet-5, VGG16, ResNet 101, Inception v3 and InceptionResNet V2. The morphological neural network in the experiment including the Morphological Operation Model and the Adaptive MNN. Considering there are not only one type of Morphological Operation Model, only the highest classification accuracy is recorded in Table 5.5. Table 5.5 shows the comparison experimental results between CNN and MNN.

Table 5.5 Comparison Experimental Results Between CNN and MNN.

	Morphological Operation Model	Adaptive MNN	LeNet-5	VGG16	ResNet-50	Inception v3	Inception ResNet V2
Bee-Wing	87.93%	86.35%	87.78%	17.74%	86.54%	87.16%	87.72%
Augmented Bee-Wing	89.77%	90.33%	89.97%	88.7%	89.34%	91.46%	90.91%
Brain Tumor	95.33%	96.47%	90.17%	95.69%	96.30%	97.61%	97.91%
MNIST	98.93%	97.33%	98.10%	98.50%	98.79%	99.13%	99.65%
GTSRB	97.48%	97.53%	90.49%	95.32%	97.39%	97.89%	98.01%
Chest X-Ray	96.75%	98.75%	92.40%	94,89%	97.04%	98.63%	98.78%
COVID-19	96.57%	97.33%	93.96%	94.91%	95.68%	97.09%	97.92%
Cat & Dog	78.31%	78.64%	96.00%	97.53%	98.32%	99.62%	99.83%
SCGS	97.75%	98.14%	90.97%	93.18%	94.15%	97.96%	97.05%

Table 5.5 shows the performance of seven deep learning model. These seven models can also be classified as two categories: the morphological neural networks and the

convolutional neural networks. The two kinds of deep learning models are based on different feature extraction mechanisms, the mathematical morphology and the mathematical convolution, respectively.

In the ecology datasets and medical datasets: the Bee Wing Dataset, the Augmented Bee Wing Dataset, the Chest X-Ray Dataset and the COVID-19 Dataset. The features in these samples are relatively easy to tell. The performance of the MNNs and the CNN are similar, which indicate both of the models can extract enough features. However, considering the LeNet-5 and the Morphological Operation Model both contains two feature extraction layers and CNN requires more, the MNN could use less parameters to achieve a similar and even better performance. The following results show MNN is can be applied to image smoothing and feature extraction in ecology dataset and medical datasets.

In the recognition tasks, such as digital recognition, shape recognition and traffic sign recognition. MNN and CNN also can achieve similar results, while MNN can still use less parameter than CNN. The experimental results in MNIST Dataset, Traffic Sign Datasets and Traffic Sign Dataset, shows MNN is good at shape recognition and contour extraction.

In a more general image classification task, such as the Cat VS Dog Dataset, the experimental result shows MNN has a limitation to recognize more detailed features. Since dogs and cats shares a very close features, such as noses, eyes and ears, the MNN performs poor and achieves almost 20% lower accuracy. The reason is MNN has troubles in extracting features which has similar feature and shapes. However, the CNN models are fundamentally designed for this Dog VS. Cat recognition task.

In conclusion, the MNNs are designed based on mathematical morphology and it is good at shape representation, contour recognition and image smoothing. Compared with CNN model, MNN's limitation is it cannot recognize objects with similar features, such as whether an object is a Dog or Cat. To overcome this limitation in MNN, a new feature extraction layer is proposed in the next chapter.

5.3 The Attention Morphological and Convolutional Neural Network

In Section 5.2, experimental results show the MNN is able to achieve a relatively high performance in image smoothing, shape recognition and contour extraction with a relatively small parameters with CNN. And CNNs are able to be applied to images which share some similar features but with more feature extraction layers. Based on the following experimental results, a novel feature extraction layer which combines both the advantages of convolution layer and morphological layer is proposed in this section.

The attention MCNN layer's structure contains three parts: The Convolution layers, the morphological layers and an attention module. In the feature extraction layer, each feature map has the same size. The convolutional layers perform the convolutional operation while the morphological layers perform the morphological operation. The attention module is applied to calculate the weights of each layer, including all the convolutional layers and morphological layers. The purpose in this design is to weight each layer and make the model to achieve the best performance. Figure 5.3 shows the proposed Attention MCNN for feature extraction layer and Table 5.6 shows the technical detail of the design in the proposed structure.

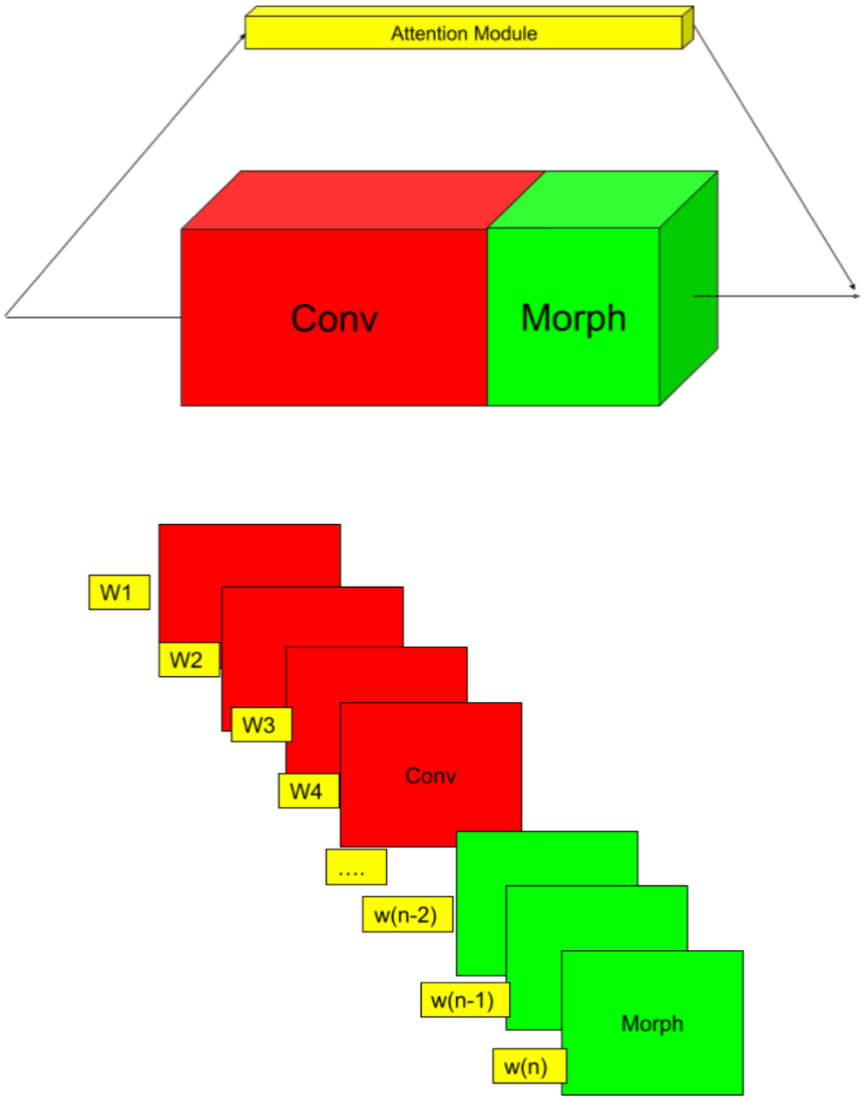


Figure 5.3 The Attention MCNN Extraction Layer and Feature Maps. The upper part shows the Attention MCNN Extraction Layer and the lower part shows the organization of feature maps.

Table 5.6 The Technical Detail in the Proposed Structure

No. of Filters in each feature extraction layer	Convolutional Layers in CNN	Morphological Layer in MNN	Attention MCNN layer In MCNN
Structure 1	32	4	10 Conv + 4 Morph
Structure 2	64	4	15 Conv+ 4 Morph
Structure 3	128	4	30 Conv+ 4 Morph
Structure 4	312	4	60 Conv+ 4 Morph
Structure 5	624	4	100 Conv + 4 Morph

The second Colum of Table 5.6 shows the common filter numbers in CNN extraction layer, the third column shows the filter numbers in MNN and the fourth column shows the proposed filter numbers in the MCNN feature extractor. Although morphological layers only contain 4 layers in each feature extraction layer, the attention module could train a learnable weight for each layer and the convolutional layers also reduced tremendously compared with the reverent CNN layers. To evaluate the performance of the proposed feature extraction structure, the CNN models are used as a baseline model and reverent convolutional layers are replace to Attention MCNN layers.

The new model with MCNN layers is named the MCNN model and Table 5.7 shows the experimental results for MCNN model in the ecology datasets and medical datasets and other datasets that have been mentioned in this research.

Table 5.7 The Experimental Results for MCNN Model

	MCNN	Morphological Operation Model	Adaptive MNN	LeNet-5	VGG16
Bee-Wing	87.17%	87.93%	86.35%	87.78%	17.74%
Augmented Bee-Wing	92.03%	89.77%	90.33%	89.97%	88.7%
Brain Tumor	96.79%	95.33%	96.47%	90.17%	95.69%
MNIST	98.95%	98.93%	97.33%	98.10%	98.50%
Traffic Sign	97.44%	97.48%	97.53%	90.49%	95.32%
Chest X-Ray	97.99%	96.75%	98.75%	92.40%	94.89%
COVID-19	97.01%	96.57%	97.33%	93.96%	94.91%
Cat & Dog	98.75%	78.31%	78.64%	96.00%	97.53%
GTSRB	98.97%	97.75%	98.14%	90.97%	93.18%

In. Chapter 4, a joint task learning model is mentioned and applied to chest X-ray ‘s classification and localization task. Based on the MCNN layer, a new joint learning model using MCNN layer is applied. Table 5.8 shows the experimental results of the new model’s performance.

Table 5.7 The Experimental Results for MCNN Model

Model	Classification Accuracy	Segmentation MAP
VGG16	89.27%	58.45%
MNN+ VGG16	94.14%	60.73%
CBAM + VGG16	93.85%	71.78%
MNN+CBAM+VGG16	90.85%	63.85%
MBAM+CBAM+VGG16	95.73%	78.72%
MCNN	96.47%	80.36%

The proposed deep learning model use MCNN layer. Compared to CNN models, the proposed model can utilize less convolutional layers in the feature extraction and achieve a relative higher test accuracy in different tasks. Compared to MNN model and CNN, the MCNN model is able to utilize both advantages of MNN and CNN. And also overcome the difficulties in MNN.

5.4 Conclusion

This chapter discussed more about how morphological neural network performs on the ecology dataset and the medical dataset. It can be described as three parts:

First, then MNN are used on the Bee Wing datasets. The experimental result shows the MNNs can performs similar results than CNN, but with a small parameter in the feature extraction layers in the bee wing datasets. It proves MNN is also useful in the bee wing classification task.

Second, the MNNs are applied to more dataset such as the Brain Tumor Dataset [48], the MNIST Dataset [49], the Traffic Sign dataset [50], the Geometric Shapes Dataset and the Cat and Dog dataset [51]. The purpose in these experiments is to explore the boundary for MNNs. The experimental results in as the Brain Tumor Dataset [48], the MNIST Dataset [49], the Traffic Sign dataset [50], the Geometric Shapes Dataset proves it can be useful in contour extraction, shape representation and image smoothing. But the results in the Cat VS Dog dataset shows the MNN is hardly to recognize items with similar features, such as the dog and cats all contains legs, ears and nose. Since these features are

hard to extract and analysis in the MNN, it requires MNN to combine some convolutional layers in the model.

Third, a feature extraction layer is developed, which combines both the morphological layer and the convolutional layer. In the proposed feature extraction structure, contains 4 adaptive morphological layers and different numbers of convolutional layers. All layers concatenated with the same shape by an attention module. The attention module is used to weight each layer, convolutional or morphological. The weight is learned in the training process with a random initialization. With the MCNN layer, a MCNN model, similar with VGG16 structure, but replaced by the MCNN layers, rather than the convolutional layers are developed. Experimental results shows the proposed MCNN model can achieves a better results than CNN or MNN in all datasets which has been mentioned in this research.

REFERENCES

[1] LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series." *The handbook of brain theory and neural networks,* 3361(10):193-202. 1995.

[2] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Network", *Conference on Neural Information Processing Systems* (*NIPS*), 2012

[3] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2015.

[4] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* , 2014.

[5] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[6] Huang, Gao, et al. "Densely connected convolutional networks." *arXiv preprint arXiv:1608.06993* , 2016

[7] Karpathy, Andrej, et al. "Large-scale video classification with convolutional neural networks." *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition.* 2014.

[8] Potapova, Rodmonga, and Denis Gordeev. "Detecting state of aggression in sentences using CNN." *arXiv preprint arXiv:1604.06650,* 2016.

[9] Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." *Nature 529.7587* (2016): 484-489.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde, Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, "Generative Adversarial Networks", *Machine Learning arXiv:1406.2661v1,* 2014

[11] Thessen AE. (2016) Adoption of machine learning techniques in Ecology and Earth Science.*PeerJ,PrePrints 4:e1720v1 https://doi.org/10.7287/peerj.preprints.1720v 1*

[12] Bell J, "Tree-based methods". *In Fielding AH(Ed.) Machine Learning, Methods for Ecological Applications. Springer US, New York,* 89-106 pp. 1999

[13] Cutler, D. Richard, et al. "Random forests for classification in ecology." *Ecology 88.11* (2007): 2783-2792.

[14] Boddy L, Morris C, "Artificial neural networks for pattern recognition", *In Fielding AH(Ed.) Machine Learning, Methods for Ecological Applications. Springer US, New York,* 37-88 pp.

[15] Ben-Hur A, Horn D, Siegelmann HT, Vapnik V, "Support vector clustering", Journal of Machine Learning Research 2: 125-137. (2001)

[16] Chen DG, Hargreaves NB, Ware DM, Liu Y, “A fuzzy logic model with genetic algorithm for analyzing fish stock-recruitment relationships”, *Can. J Fish. Aquat.Sci.57(9)* 1878-1887 (2000)

[17] Silva, Felipe “Automated Bee Species Identification Through Wing Images” *Ecological Informatics*, 2014.

[18] JOHN, G.; LANGLEY, P. Estimating continuous distributions in Bayesian Classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, p. 338–345, 1995.

[19] SHINN, A.; KAY, J.; SOMMERVILLE, C. *The use of statistical classifiers for the discrimination of species of the genus gyrodactylus (monogenea) parasitizing salmonids. Parasitology*, v. 120, n. 3, p. 261–269, 2000.

[20] WITTEN, I. H.; FRANK, E.; HALL, M. A. Data Mining: *Practical Machine Learning Tools and Techniques. 3. ed. Amsterdam: Morgan Kaufmann*, 2011.

[21] Stefan Schneider, Graham W. Taylory, Stefan C. Kremer, “Deep Learning Object Detection Methods for Ecological Camera Trap Data”, *arXiv:1803.10842v1 [cs.CV]* , 2018

[22] H. C. Shin et al., "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning", *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285-1298, May 2016.

[23] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna, “Rethinking the Inception Architecture for Computer Vision”, *arXiv:1512.00567, 2015*

[24] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi, Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, *arXiv:1602.07261v2 [cs.CV]*

[25] Kingma, Diederik, and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980*, 2014.

[26] Marcus D. Bloice, Christof Stocker, and Andreas Holzinger, “Augmentor: An Image Augmentation Library for Machine Learning”, *arXiv preprint arXiv:1708.04680*

[28] West, Jeremy; Ventura, Dan; Warnick, Sean (2007). "Spring Research Presentation: A Theoretical Foundation for Inductive Transfer", *Brigham Young University, College of Physical and Mathematical Sciences*.

[29] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, Advances in Neural Information Processing Systems 25, NIPS (2012)

[30] Jason Yosinski, Jeff Clune, Yoshua Bengio, Hod Lipson, “How transferable are features in deep neural networks?”, *Advances in Neural Information Processing Systems 27*, pages 3320-3328. Dec. 2014, arXiv:1411.1792 [cs.LG]

[31] Sergey Ioffe SIOFFE, Christian Szegedy "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift" , arXiv:1502.03167 [cs.LG]

[32] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi"Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning", *arXiv:1602.07261 [cs.CV]*

[33] F.Y. Shih and O.R. Mitchell, ''Threshold decomposition of grayscale morphology into binary morphology,'' *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 1, pp. 31-42, Jan. 1989.

[34] R.M. Haralick, S.R. Sternberg and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 4, pp. 532-550, July 1987.

[35] F.Y. Shih, *Image Processing and Mathematical Morphology: Fundamentals and Applications, Taylor & Francis Group, CRC Press, Boca Raton, FL*, 2009.

[36] F.Y. Shih and J. Moh, "Implementing morphological operations using programmable neural networks," *Pattern Recognition*, vol. 25, no. 1, pp. 89-99, Jan. 1992.

[37] J.L. Davidson and F. Hummer, "Morphology neural networks: An introduction with applications," *Circuits Systems and Signal Process*, vol. 12, pp. 177-210, June 1993.

[38] J. Masci, J. Angulo, and J. Schmidhuber, "A learning framework for morphological operators using counter-harmonic mean," *Proceedings of 11th Int. Symp. Mathematical Morphology: Its Appl. Signal Image Process., Springer, Berlin, Heidelberg*, pp. 329-340, 2013.

[39] F.Y. Shih, Y. Shen, and X. Zhong, "Development of deep learning framework for mathematical morphology," *Int. J. Pattern Recognit. Artificial Intell*, vol. 33, no. 6, p. 1954024, June 2019

[40] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature, vol. 521, pp. 436-444*, May 2015.

[41] X. Zhou, R. Takayama, S. Wang, T. Hara, and H. Fujita, "Deep learning of the sectional appearances of 3d CT images for anatomical structure segmentation based on an FCN voting method," *Medical Physics*, 2017.

[42] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: visual explanations from deep networks via gradient-based localization," *Proc. Intl. Conf. Computer Vision*, 2017

[43] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," *Proceedings of Intl. Conf. Computer Vision and Pattern Recognition*, 2016.

[44] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, and T. S. Chua, “Sca-cnn: spatial and channel-wise attention in convolutional networks for image captioning,” *Proceedings of* Intl. Conf. Computer Vision and Pattern Recognition, 2017

[45] F. Y. Shih, Y. Shen, and X. Zhong, “Development of deep learning framework for mathematical morphology,” Intl. Journal Pattern Recognition and Artificial Intelligence, vol. 33, no. 6, p. 1954024, June 2019.

[46] R. Kotikalapudi, et al., “Keras-vis,” https://github.com/raghakot/keras-vis (Accessed 05 June 2018).

[47]Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L. Yuille, “DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, *arXiv:1706.05587v3 [cs.CV]*

[48] J. Cheng, “Brain tumor segmentation using holistically nested neural networks in MRI images”, *The International Journal of Medical Physics Research and Practice*, July 2017

[49] THE MNIST DATABASE of handwritten digits". Yann LeCun, Courant Institute, NYU Corinna Cortes, Google Labs, New York Christopher J.C. Burges, Microsoft Research, Redmond.

[50] J. Stallkamp, M. Schlipsing, J. Salmen and C. Igel, "The German Traffic Sign Recognition Benchmark: A multi-class classification competition," *Proceeding of Int. Joint Conf. Neural Network*, San Jose, CA, 2011, pp. 1453-1460.

[51] Bang Liu, Yan Liu, Kai Zhou “Image Classification for Dogs and Cats”, *International Research Journal of Engineering and Technology (IRJET)*, December 2019.